Decentralised Compliant Control for Hexapod Robots: A Stick Insect Based Walking Model

Hugo Leonardo Rosano-Matchain

Doctor of Philosophy
Institute of Perception, Action and Behaviour
School of Informatics
University of Edinburgh
2007

Abstract

This thesis aims to transfer knowledge from insect biology into a hexapod walking robot. The similarity of the robot model to the biological target allows the testing of hypotheses regarding control and behavioural strategies in the insect. Therefore, this thesis supports biorobotic research by demonstrating that robotic implementations are improved by using biological strategies and these models can be used to understand biological systems. Specifically, this thesis addresses two central problems in hexapod walking control: the single leg control mechanism and its control variables; and the different roles of the front, middle and hind legs that allow a decentralised architecture to co-ordinate complex behavioural tasks. To investigate these problems, behavioural studies on insect curve walking were combined with quantitative simulations.

Behavioural experiments were designed to explore the control of turns of freely walking stick insects, *Carausius morosus*, toward a visual target. A program for insect tracking and kinematic analysis of observed motion was developed. The results demonstrate that the front legs are responsible for most of the body trajectory. Nonetheless, to replicate insect walking behaviour it is necessary for all legs to contribute with specific roles. Additionally, statistics on leg stepping show that middle and hind legs continuously influence each other. This cannot be explained by previous models that heavily depend on positive feedback controllers. After careful analysis, it was found that the hind legs could actively rotate the body while the middle legs move to the inside of the curve, tangentially to the body axis.

The single leg controller is known to be independent from other legs but still capable of mechanical synchronisation. To explain this behaviour positive feedback controllers have been proposed. This mechanism works for the closed kinematic chain problem, but has complications when implemented in a dynamic model. Furthermore, neurophysiological data indicate that legs always respond to disturbances as a negative feedback controller. Additional experimental data presented herein indicates that legs continuously oppose forces created by other legs. This thesis proposes a model that has a velocity positive feedback control modulated via a subordination variable in cascade with a position negative feedback mechanism as the core controller. This allows legs to oppose external and internal forces without compromising inter-leg collaboration for walking. The single leg controller is implemented using a distributed artificial neural network. This network was trained with a wider range of movement to that so far found in the simulation model. The controller implemented with a plausible biological

model further increase the connection with the real insect. Further similarities with the stick insect in support of this controller are presented.

The control hypotheses and behavioural results were incorporated into a 3D dynamic robot simulation. The simulation can replicate the turns made by the stick insect more precisely than any previous model. Results demonstrate that the single leg controller can operate in a dynamic system by opposing external forces. Simultaneously, the controller can be integrated in a decentralised architecture and still co-operate with other leg controllers. The robot simulation was tested at various surface inclinations and with variations in weight and size. Evidence is presented that indicates the feasibility of implementing this model in a real robot.

Acknowledgements

First, I would like to thank my supervisor Dr. Barbara Webb for her experienced guidance, suggestions and patience during this research; she has invaluably improved this research. Additionally, I have to thank her for her help on non-academic matters as well, which helped me focus on my work.

My research could not have been possible without the financial support of CONACyT México, which continuously sponsored this thesis for almost four years. Many thanks to my second supervisor Mark Van Rosum, for our discussions at various stages of this thesis. To Russell Smith, primary author of ODE, I thank him for sharing and supporting the physics dynamic libraries used herein.

Specials thanks must go to the Biological Cybernetics Institute at the University of Bielefeld. To Prof. Dr. Holk Cruse, Prof. Dr. Josef Schmitz, Dr. Volker Dürr and all the PhD students that with their comments and suggestions, put this thesis on the right track. In addition, I would like to thank the Neural Control of Locomotion Group at the University of Cologne, to all the staff and students; particularly Andreas Büschges for his orientation in the early stages of this research.

Thanks to the Edinburgh Butterfly and Insect World, for providing the first stick insects that would eventually father all the insects used for this thesis. Particularly, I would like to thank Kevin for sharing his expertise on stick insect care and breeding.

I am grateful with those that help me with coding, proofreads and invaluable feedback. Thanks to Adrian Haith, Mikel Kochenderfer, Tim Lukins, Michael Mangan, Mark Payne, Darren Smith, Matt Szenher, Matt Whitaker and Jan Wessnitzer. Many more should be included and I apologise for not having enough space. I would also like to thank all the people at the IPAB, the badminton squad, the IAR team and the 'ipub' members for making this a brilliant experience.

I would like to thank my parents for providing me with the education, support and values that allowed me to pursue this degree. Thanks also to all my family and friends that were constantly monitoring my progress and encouraging me to do my best.

Last, but by no means least, I owe immense gratitude to my wife, Fatme; more than I could ever express. She has always been there to help and support me in good and bad times. Specially, I have to thank her for her enormous patience during all the weekends and holidays I had to work. But most importantly, for all this time I have been without a proper job.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Hugo Leonardo Rosano-Matchain*)

Table of Contents

List of Figures

Chapter 1

Introduction

The work described in this thesis aims to improve the performance of legged robotic locomotion. To achieve that, it was decided to study how nature solves the problem in the first place. This objective and approach raise the following questions: Why use nature as inspiration? Why go as far as to understand nature? Why not solve the problem directly? Why use legs in the first place? The purpose of this chapter is to try to focus these questions; and in the process put forward the specific questions this thesis aims to answer.

1.1 Legs vs Wheels

One question at the heart of this thesis is why use legs? Although legged robots have improved over the years, wheeled vehicles are still preferred for many applications, particularly for commerce and industry. That is because the goal is not to get a vehicle to move around *per se*, but to solve some specific task. Therefore, the use of reliable but "non-complicated" machinery is preferred. The main advantage for wheeled vehicles is their continuous stability, which can be achieved without any control algorithm. However, there are two key reasons to support the use of legged walking machines: legs require only non-continuous areas of solid ground for moving (Raibert, 1986) and navigation is not constrained.

Not needing continuous support areas has remarkable consequences since it applies to all position variables in 3D space, $\{\hat{x}, \hat{y}, \hat{z}\}$. Legged systems can overcome vertical discontinuities in a similar manner to horizontal. Wheeled vehicles are particularly less capable of dealing with discontinuities in $\hat{z}$. In reality, all surfaces are non-continuous and the evaluation of motion depends on the relation between gap dis-

tance and wheel size. However, although one can increase the size of the wheel, the size cannot be further changed while in motion, i.e., wheel size is not variable. Consequently, if the gap/radius relation is large, the vehicle will be going up and down at each step. Motion would be very inefficient in energy consumption and unpleasant if one is on board[1]. However, the real disadvantage in terms of requiring continuous surfaces for locomotion is when travelling vertically. It is far more common to find vertical discontinuities than horizontal ones and sometimes is not only inefficient, but also impossible to overcome them.

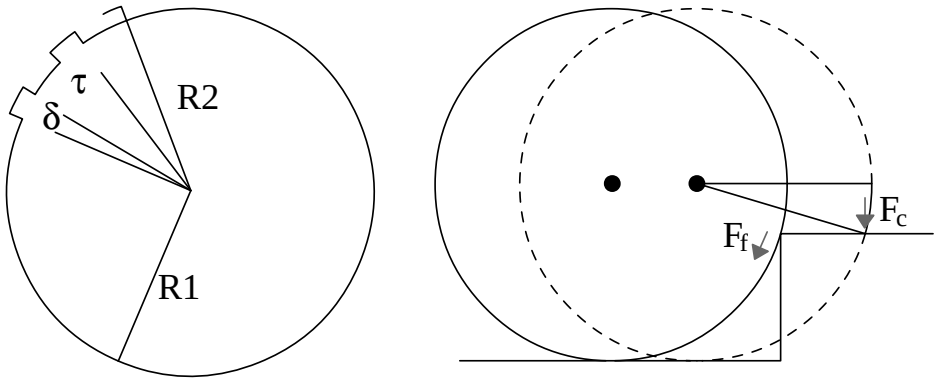


Figure 1.1: General wheel model.

Left: R2 is the radius to the most distant feature, R1 is the radius to the main wheel body, τ is a periodicity variable, and δ is the feature size. Right: two type of forces found on two different wheel morphologies, one is a purely frictional force (F_f), and the other is a contact force (F_c).

Most wheels can be described by three important parameters: the depth of their dentations (R2-R1); the angular size of the dentations δ; and their frequency $1/\tau$, shown on the left hand side of Figure 1.1. Most wheels can be described by $R2 \approx R1$ and $\tau \ll 1$, which means that they depend on friction for going up a step. The larger the relationship between edge height *h* and radius *r*, the more dependent on frictional forces the lift will be, $F_f \approx h/r$. Dentations improve holding by moving the support point farther away from the edge and converting frictional forces (statistical) into contact vertical forces (constant). By controlling these parameters it is possible to increase wheel efficiency, for instance, the Whegs robot have $R2 \gg R1$ and $\tau \sim 2\pi/3$ (Quinn

[1]The amplitude is $A = r(1-((1+a/2)(1-a/2))^{1/2})$, where r is the wheel radius and the gap size is g=a r.

et al., 2002). This wheel geometry is shown in Figure 1.3.

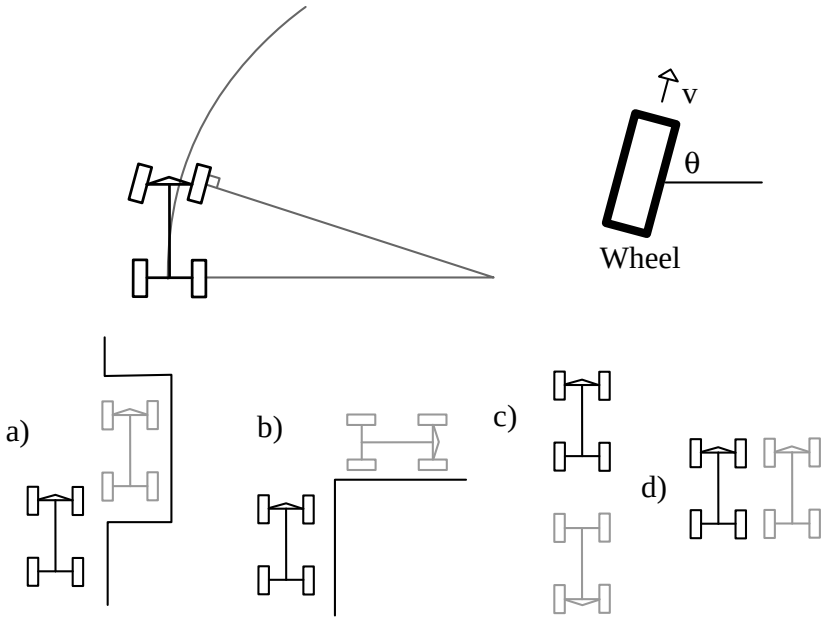


Figure 1.2: Nonholonomic motion

Wheeled vehicles are nonholonomic because they control fewer variables than those defining their position. These imply a nonholonomic vehicle's position is path dependent. Navigation paradigms are shown in examples a–d; these require complex path planning.

The second problem mentioned above is that legs do not restrict path navigation. Vehicles are usually described by position variables plus heading direction, i.e. where they are, $\{\hat{x}, \hat{y}, \hat{z}\}$ and their orientation, $\hat{\theta}$. The problem is that wheeled vehicles only control two variables, velocity and wheel direction. Motion in a N dimensional space by controlling (N-m) variables is known as a nonholonomic system. It is in other words, path-dependent. The problem comes down to the single wheel, which is controlled only by two variables, i.e, angular speed and wheel direction. It is still possible to navigate the whole of the space, however, it takes longer and requires advance navigation planning.

For instance, in Figure 1.2 a simple vehicle is shown turning to the right. Note that it is not theoretically so different to have just one wheel; the main difference being the local limit of wheel direction. In the figure, four representative problems are presented, the vehicle with black lines is the initial position and the grey is the

desired position. The first thing to note is that the vehicle requires more space to move around and position itself in the correct place and direction. Consequently, the time required to get there increases with the complexity of the problem. Suppose in Figure 1.2(a) the space available for the final position is just about the right size, it would require many repetitive steps back and forward various directions to eventually move to the right. Even the single wheel needs two time dependent movements to move to a position parallel to its current direction (Figure 1.2(d)). Jointed legs on the other hand are capable of moving to any direction without previous adjustments. With no height variations, legs can be virtually represented by spheres in a socket joint. In addition, it is possible to take the analogy even further and include variability in height, which means that the virtual spheres are adaptable in size. Imagine now solving the problems shown in Figure 1.2 with the vehicle standing on radii variable spheres. The possibilities are endless, and planning is very straightforward.

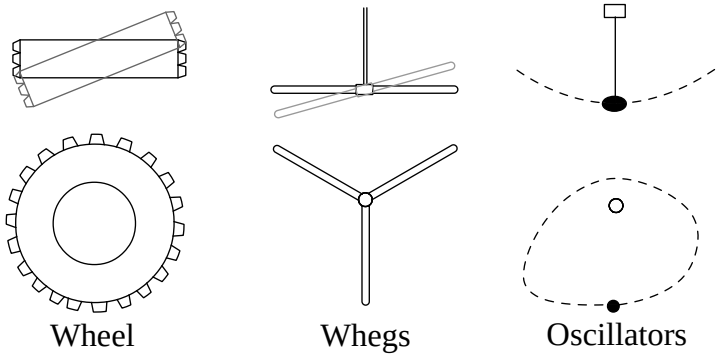


Figure 1.3: Wheels and simplified legs share disadvantages

Simplified legs improve motion for vertical discontinuities similar to wheels for rough terrain. However, simple legs pose no navigational advantage to traditional wheels. Legs with fixed trajectories, either mechanical (Whegs) or pre-programmed (oscillators) have a resulting reduction in degrees of freedom. All shown in this figure classify the system as nonholonomic.

The distinction between wheels and legs can be very ambiguous if one considers only the shape of the appendage. It is important to remember that the advantage of using legs is not only because of morphology. The functionality of wheels is intrinsically nonholonomic because all wheels are driven by two rotating axes; one for controlling speed and the other for controlling direction. Jointed legs, on the other hand, are con-

trolled by motors that do not specialise in any particular dimension. Therefore, have the potential to overcome the nonholonomic navigation problem if all degrees of freedom are exploited. Therefore, the functionality of wheels or legs is not only based on their morphology and it is important to identify weaknesses and strengths when it comes to choose vehicle platforms. For instance, on the middle and right hand side of Figure 1.3, the appendages' movements are so simple that they could easily be interpreted as wheels. Furthermore, based on their operation, both are nonholonomic, just as for the wheel on the left.

1.2 Using Legs

Legged robots are more flexible and their navigation is almost unrestricted. Ironically, that is also their main disadvantage, because they require more sophisticated control algorithms. The first detailed study on legged locomotion was done in 1887 by Eadweard Muybridge. The study was based on a series of consecutive photographs that showed various animal gaits (this was done before motion picture cameras). The first attempts to construct controllable joined segments were purely mechanical, which further restricted and complicated the problem. For instance, Adolf Ehrlich (1928) proposed a mechanical leg-propelled vehicle, which although impressive at first sight, showed no benefit in use. Nonetheless, the use of computer control in vehicles has steadily increased and nowadays almost all commercial automobiles need computers onboard as well. Dependency on control algorithms is no longer restricted to legged locomotion.

Legged robots can be divided into two main categories: dynamically stable and statically stable. The former requires continuous adjustments to posture and inertia for not falling, e.g., bipeds. The stability of the latter is not time dependent and therefore does not require real time controllability, i.e. if the entire system suddenly shuts down it will not fall. Counterintuitively, robots are at present more fragile than biological systems, even if their materials are harder than those found in nature. One of the reasons is that the energy absorbed on shock contact with the ground is not damped. In addition, the relative weight compared to their scale is usually high. Therefore, risk of falling cannot be neglected. Statically stable robots have the advantage of being intrinsically safer, falling down is typically within their own standing area, contrary to other legged systems with fewer legs that usually fall to the front or to the side. Additionally, their body is usually orientated horizontally, hence, they fall from less height relative to their body length. Dynamically stable systems are more suitable for

fast locomotion. Provided the legs are built with spring-like mechanisms motion at high speed can be very efficient. However, energy saving mechanisms for running are not restricted to biped or quadruped locomotion. The same principles of operation have also been found in fast hexapods like cockroaches. Statically stable systems require at least three contact points at all times, which means that at least four legs are needed. That increases the redundancy of the system and requires synchronous coordination between legs in contact with the ground. Therefore, statically stable systems are safer; they could have energy saving mechanisms that allow fast locomotion; but they still pose difficulties to control them, particularly if one is not to implement simplistic legs that provide no advantage over wheels.

Legged locomotion controllers can be analysed at three different levels: body trajectory, inter-joint coordination and leg coordination. The first is usually the intended direction and speed required for the vehicle, which is normally a higher-level command. Body motion is represented by six variables, three for position and three for orientation. By knowing where the body should be with respect to the feet it is possible to calculate leg movements and hence joint positioning. However, legs need to be coordinated for repositioning once they have reached an extreme position in their motion. This extreme position is usually in a posterior position; hence, it is called the posterior extreme position (PEP). Nonetheless, legs cannot be lifted carelessly because the robot could fall, e.g., if all legs on one side are lifted. When legs are lifted and moved forward to be placed down again this is known as the swing phase and while on the ground, stroke or stance phase. Forward walking requires the legs to be placed in the front again; hence, that position is known as the anterior extreme position (AEP). However, as mentioned earlier, legs are multidirectional and hence the AEP and PEP could be to the right or to the left as well as anterior or posterior.

1.3 Insect Based Robots

Computers have already been proven to solve certain problems faster and more accurately than the human brain. However, so far even the simplest walking animals are far more agile and stable than any robot built. It would be ideal to understand how biological systems operate in order to implement what is useful for robots. Animal's agility is partly due to their complex morphology, and this can be imitated to some extent. For instance, the many degrees of freedom or the geometrical arrangement can be replicated. However, sensory integration, decision-making and motor control are very

complex as well. These are responsible for most of the animals' successful locomotion. Unfortunately, at present more problems focus on understanding the information processing within the nervous system. Nowadays there is still debate about how biological neural networks operate. The nervous system operation is not easily understood because it does not only depend on its topology, dynamic parameters inside each neuron are strongly temporally dependent.

On the other hand, just as one can use biological systems as inspiration or target systems, one could also use robots as platforms for testing biological hypotheses (Büschges, 2004), provided both systems operate with similar rules, analogous information, and a proper validation is performed (Webb, 2006). However, even if neuron interconnections and their operation were known, the dynamics are so complex and seemingly stochastic that it is not possible to obtain analytical solutions. Richard Feynman (1979) said that, "the rules that make nature work are simple", however, the difficulties of prediction lie on "the multiplicity of its actions and interconnections"; hence, empirical and numerical approximations are required. Just as neuroscience requires numerical methods for analysis, neuroethology requires biorobotics, because the latter requires to be embedded in an environment, i.e., the agent is situated. This thesis supports the idea that biorobotics is not only useful, but also currently necessary for the study of neuroethology and therefore to the full understanding of neuroscience.

Locomotion is a complicated task that involves sensory neurons, interneuron processing and motoneurons. In addition, information related to leg motion is influenced by other body activities and brain commands. Ideally, it would be better to have a system whose walking controller is independent of other signals. In insects, the nervous system controlling walking is divided into four sections, the most sophisticated in the head and three neural concentrations for each pair of legs, named ganglia[2]. The head is known to signal higher-level commands like walking intention and directionality. However, it does not have substantial role for dedicated joint control (Bässler, 1988; Bässler et al., 1985). Each ganglion is approximately 1 mm across in *Carausius morosus* and has fewer neurons than the brain. Nevertheless, insect locomotion is still more efficient, robust and flexible than that of any current robot. With the aim of understanding the motor and behavioural control performed by animals at an individual neuron level, insects in general are chosen. Their relatively simple neural architecture allows easier electrophysiological recordings and recognition of individual neurones. Furthermore, during experiments, insects can be manipulated genetically and morphologically

[2] A group of nerve cells forming a nerve centre.

more easily than other animals, such as mammals (Full, 1997).

1.4 Complications

The first constraint when working with insects is the difference in size between the actual insect and the proposed robot. Effects of gravity are not a problem for insects as they are for robots, for instance, leg inertia. Furthermore, because of the relation between weight and muscle power, insects can lift several times their own body weight without damaging their joints; allowing them to take a larger variety of postures. Furthermore, claws and other adhesive structures on their feet allow them to walk on walls and ceilings. Positions achieved by these means are not considered for this research, as they are not easily implemented in a robot. However, advantages of using legs discussed earlier did not consider having claws, e.g., holonomic navigation. Therefore, it is desirable to have legs even if they lack claws.

Insects have provided various simple solutions for the control of legs in robotics. Work has been done on simplifying the three control levels mentioned in section 1.2: body trajectory, inter-joint coordination and leg coordination. Unfortunately, sometimes in the process legs have lost the strongest contributions to locomotion mentioned above. Fixing leg trajectories or using less than three degrees of freedom is one of the first steps towards “wheeling” legged vehicles, i.e., modifying legs to behave more like wheels instead of exploiting their advantages. Furthermore, it might be very difficult to have flexible leg coordination with legs following fixed trajectories, as the former depends on the latter.

There have been hexapod robots capable of moving in various directions while avoiding obstacles and correcting for feet misplacement. However, although perhaps initially inspired by biological systems, some became too different to be further used to test biological hypotheses. Imitation of movement is not sufficient to suggest biological functionality, particularly if the ‘imitation’ is not initially based on what is already known. Ultimately, such walking controllers cannot be further improved based on what biologists have to offer because low-level implementations are not compatible. For instance, centrally controlled architectures cannot implement single leg controller strategies. Essentially, following a biologically based approach and then departing from it prematurely imply that the target animal do not have any more to offer, which is an idea not easily supported. This thesis intends to improve robot hexapod locomotion towards a targeted biological system, which if eventually equalled will not only allow

a robot to overcome challenging navigational tasks, but will also enable the study of the insect's neurophysiology.

In the pursuit of achieving what animals can do, it is important to note that behaviours are found to be complex when the environment and input to the system are complex as well. For instance, repetitive movements could result from a monotonous control signal, or from a potentially complex internal control variable responding monotonously to an invariable stimulus. Even human movements can be accurately described if the task is not demanding, which by no means implies that the underlying mechanisms are simple. Unfortunately, insect's simpler processing of the world tends to increase the chances of observing such monotonous or repetitive behaviours.

Some animals are also more prone to simplify their behaviour because of their survival strategies. Cockroaches rely on central pattern generators[3] (CPG) for running because at high speed there is no time to process sensory information. Their body and legs have adapted over time to travel at high speeds and still be very stable. Submarine wildlife do not require leg positioning for swimming propulsion and therefore CPGs are used by many aquatic animals. However, cockroaches walking at non-panic speeds and crustaceans turning on the ground show a very different behaviour. It is possible that these animals have different pre-programmed behaviour for each situation. However, results instead suggest contextual neural modulation. Ultimately, choosing a biological target depends on the problem at hand and the biological system is going to be different for each case.

Straight walking is a complicated locomotion task that requires coordination, stability, sensory processing and motor control. However, after the transient period of starting to walk, the periodicity that results from undisturbed walking obscures the need for some sensors and subsequent processing. There are still unanswered questions regarding joint and leg control for straight walking: is it not possible that straight walking is a simplification of turning? If so, it should be more insightful to solve for the more generic situation and then verify results for particular cases. The approach followed in this thesis is to study the behaviour of turning and then verify how the system responds when turning is zero, i.e., straight walking. By following this approach, stereotypic movements will be minimized and hence the influence of sensory information for walking will be highlighted.

[3]Neuronal network capable of generating a rhythmic pattern without phasic input.

1.5 Thesis Contribution

The stick insect *Carausium morosus* is one of the best biological targets for land vehicle locomotion. It is a clear example of legged-locomotion superiority on unexpected rough terrain and flexible manoeuvrability. It has been extensively studied and offers important insights into how a biological walking architecture is organized. Many robots have been constructed based on this insect, yet none has been demonstrated to perform better than the stick insect. Furthermore, neurophysiological data is very incomplete and hence, it would be desirable to have a compatible testing platform on which to test hypotheses.

This thesis aims to produce a hexapod walking control architecture that incorporates elements of stick insect locomotion that would allow it to outperform current implementations. Furthermore, particular paradigms are addressed in which leg flexibility and body manoeuvrability demonstrate superiority over wheeled vehicles. This thesis recognizes that autonomous robots must ultimately fulfil tasks other than moving about. Therefore, higher-level commands from the brain for locomotion controllability are of extreme importance.

The contributions to insect experimental and analysis methodology include a non-invasive behavioural experiment that allows precise control of the insect walking direction. Moreover, this thesis introduces an adjustment to current techniques studying body kinematics that is more suitable for legged-systems. The instant axis of rotation of the body was introduced to analyse turning. Additionally, a direct correlation between individual leg's kinematics and body motion is analysed.

The contributions to biology by this thesis include a better understanding of insect walking behaviour. This thesis demonstrates that thoracic differentiation is vital for the insect turning manoeuvrability, particularly for the control of turning. Furthermore, experimental results demonstrate a clear link between visual inputs and front leg directionality that could lead to mapping of signals from the brain to individual legs.

Analyses of behavioural experiments presented herein were combined with current information on the stick insect to propose a robotic walking control model that closes the loop between lower-level joint control and visual input. The model improves hexapod robotic locomotion by proposing a single leg controller that allows legs to comply with each other while sustaining specific commands. This allows legs to be decentralised and thus they require less explicit coordination. Furthermore, the thoracic differentiation shown by the insect improves manoeuvrability of the robot by simple

mechanisms and it is controlled by a single higher-level command. Much effort was dedicated to avoid proposing a biologically 'irrelevant' model (Webb, 2001).

As a whole, this thesis supports biorobotic research by demonstrating that robots can be improved by understanding biology and that these robotic models can be used to further understand biological systems.

1.5.1 Organization

Robotics is a multi-disciplinary area that covers different scientific and engineering fields. Terminology among these areas is standard, however, these are not always compatible with biology, which is a broad scientific area in its own. Whereas some of the terms are straightforward to match with its counterpart (camera/eye), others are not as familiar (stress gauge/chordotonal organ). This thesis does not favour any particular reader, hence, 'obvious' definitions are included where necessary.

First, information known for the stick insect that is relevant to this thesis is summarized (chapter 2). The key elements used in the work presented here are stressed. Furthermore, it is important to explore models that have been proposed to date. In the discussion section in this chapter, missing elements in these models are identified and alternatives to those problems are presented.

Chapter 3 presents an experimental setting that allows investigation of complex insect locomotion linked to visual commands. Within that chapter, data is analysed and the bases for the walking model are presented. The proposed model for the single leg is introduced in chapter 4 and its integration into the whole walking controller in presented in chapter 5. It is important to mention that the control of turning was first analysed and solved, and subsequently results for straight walking were verified. Therefore, results in chapter 5 of the robot simulation actually precede those in chapter 4. Nonetheless, it is easier to understand and present results for the controller implemented in the single leg when the robot walks straight forward.

In chapter 4 an analytical equation is introduced for the control of legs. However, this is not easily implemented based on what is currently known in neuroscience. Therefore, in chapter 6 a neural network that replaces this equation is proposed.

Conclusions of this work, discussion and future work is then presented in chapter 7.

To aid reading this document, two reference sections are also included: On page 207 the *List of Symbols* contains the variables used across chapters. The *Glossary* on page

210 contains a list of specialised words with their definition.

Chapter 2

Biological Overview and Related Work

This chapter introduces the biological target based on the following questions: What is the stick insect? What does it do? How does it do it? Once described, a discussion on robotic and kinematic models that have been proposed based on hexapod insects is presented; particularly for the stick insect and the cockroach. It is intended at the end of this chapter to present the specific problems this thesis aims to solve.

Prior to these, section 2.1 starts with an overview of the insect and its most salient features. The morphology of the insect is described in section 2.2. Of particular importance are the leg geometry and the names of joints and segments because those are referred to extensively. This section also details sensors available to the insect's nervous system and the kinematics of legs.

Behaviours can be very complex, like hunting and nest building. However, not even the basics have been understood, like standing or walking. These are important to robotics because it is more desirable to control behaviours rather than controlling legs or joints. This is summarised in section 2.3.

It is not practical to mimic behaviours without considering internal mechanisms of sensory and interneuron information processing. There are no guarantees that the proposed algorithm imitating the behaviour would be the same as the real one. The model could end up being a puppet with no relevance to the target system. In section 2.4 current knowledge of the stick insect nervous system and the elements taken for this thesis are presented.

This thesis is by no means the first attempt to implement a robot or simulation based on insects. Others fulfilled their particular objectives and thus are important to mention, but these have various strengths and weaknesses, which are discussed farther (section 2.5).

This chapter introduces most of the terminology and background information used throughout the thesis. After this chapter, sufficient information is presented for a detailed discussion on current problems yet to solve, i.e., where this research fits in. The approach proposed to analyse and subsequently solve these problems is presented in section 2.6.

2.1 Overview

Phasmatodea[1] are usually categorised as a suborder of Orthoptera[2]. Phasmatodea are also known as stick insects or leaf insects; these names refer to their ability to camouflage with vegetation. The former resemble tree branches and the latter tree leaves. Because their survival strategy is to not be seen, they have a sedate locomotion. Furthermore, they react to danger by decreasing speed instead of fleeing. Stick insect size varies among species and with every moulting. After hatching, their size is varies from $0.5mm$ after hatching to $20cm$. Their natural environment is among branches, where they can hide from predators.

The species *Carausius morosus*, also known as the Indian stick insect, has been extensively studied morphologically and neurophysiologically. Furthermore, it has inspired many legged models that in turn have contributed to a better understanding of the underlying walking mechanism. *Carausius morosus*, which from now on will be referred only as the stick insect, represents an ideal biological target for a robot. The insect uses all degrees of freedom on its legs for navigation and neurophysiological data suggests that motion is highly sensory dependent; no central CPGs control leg movement. Furthermore, compared to other insects, the stick insect's more 'careful' with the positioning of the tarsus.

Nocturnal activity means that the insect cannot rely on visual sensors as much as tactile information; furthermore, the terrain in which it lives is highly uneven and unpredictable. Consequently, locomotion cannot depend on stereotyped movements and each joint movement has to be consistent with other joints within the same leg. That would be very difficult to achieve with a repetitive internal signal and in fact there is none central pattern generator controlling joints. Walking by the stick insect is sufficiently generic and non-trivial as to contribute to the global understanding of

[1] Phasma meaning phantom.

[2] Orthoptera belong to the groups of Neoptera. Basically, this group is characterized by its leathery forewings and membranous hind wings and chewing mouthparts.

walking locomotion.

2.2 Morphology

The adult stick insect, measures between 6 and 8 cm. The body is approximately cylindrical and the three pairs of legs are positioned towards the front. The body is divided in three regions: The head, thorax and abdomen. The thorax is further divided in the prothorax (front), mesothorax (middle) and metathorax (hind). It has eight main appendages projecting out of the body, two at each thoracic segment and two on the head (antennae). The latter are not used for supporting, however they synchronise with legs while walking (Dürr, 2001). They convey information about position of support surfaces and obstacles. Figure 2.1 shows a schematic of basic morphology.

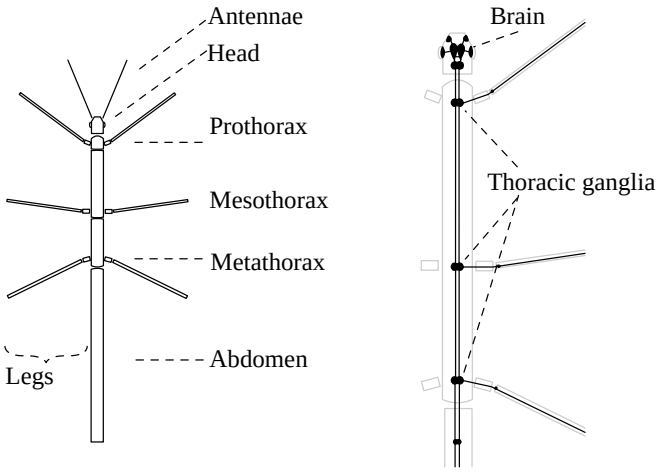


Figure 2.1: Stick insect physiology

Left: Stick insect morphology; main body segments are shown. Right: Nervous system organization. Note neural concentration at each pair of legs and highest concentration at the brain.

Stick insect legs are divided into four movable sections: the coxa, closest to the body; the trochanter-femur, or just femur as they are fused; the tibia and the tarsus. The body-coxa (BC) joint is best described as a socket joint, whereas the coxa-trochanter (CT) joint and the femur-tibia (FT) joint are hinge joints. The CT and the FT axis are parallel, hence, the femur and tibia lie on the same plane. The tarsus (foot) is further subdivided in various segments, which makes it very flexible. The tarsus also has claws

and adhesive structures to grasp and hold surfaces. It tends to move in the direction of grasping by an elastic mechanism, therefore, it only needs to be controlled in one direction (pulled).

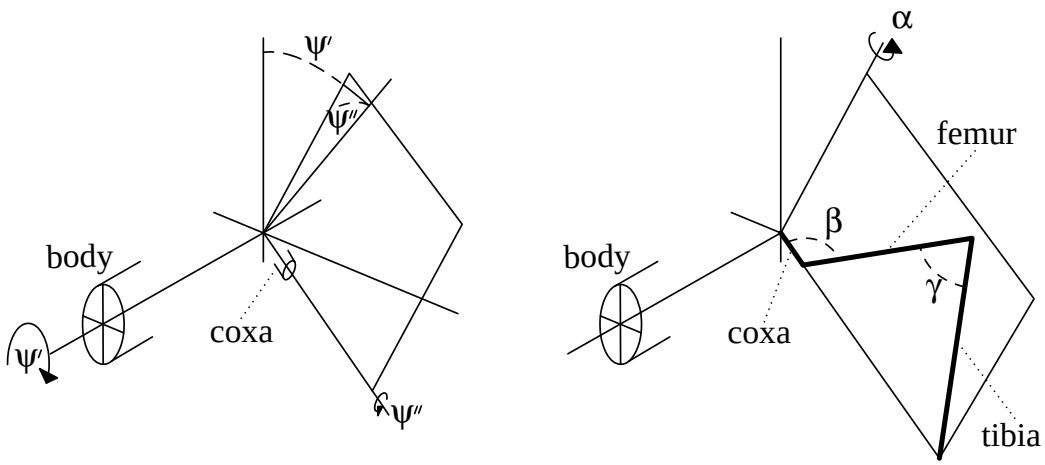


Figure 2.2: Leg orientation and variables

Left: coxa orientation relative to the body. Two rotations projects the coxa mainly downwards, first it is rotated by ψ' and then by ψ''; the first rotation is large whereas the second is small. Right: The three main angles describing joint position, α for the BC joint, β for the CT joint and γ for the FT joint. Note that the three segments are always on the plane rotated by the BC joint.

Orientation and motion of the coxa segment out of the body can be described by three orthogonal rotations, one of which is dominant. On the left hand side of Figure 2.2 are shown the two orientations that change the least, and thus both are considered fixed. However, although variability while walking is small, the angle ψ' in the stick insect is large. Consequently, the coxa segment points down instead of being parallel to the ground. On the right hand side of Figure 2.2, variables that determine most of the relative tarsus-body (TB) position are shown. The α (alpha) variable on the BC joint determines mostly how much to the front or to the back the leg plane is positioned. The β (beta) angle at the CT joint specifies how high or low the knee is (FT joint). Finally, the γ (gamma) angle determines the lateral proximity of the tarsus to the body. Note that all joints influence motion in all directions, i.e., joint variables are non-linear with respect to Cartesian coordinates. Joints and the variables used to label their angular position are used arbitrarily to name the joint. Thus, the BC joint is also referred as the

α joint, however, the actual angle is always assigned to α.

Legged terrestrial animals require interconnected solid structures to support the body, otherwise they would collapse by their own weight. Insects have all the soft tissue surrounded by a hard cover called the exoskeleton. Because the exoskeleton does not increase in size as bones do, insects have to moult as they grow. Exoskeletons have additional consequences for locomotion. In vertebrates, muscles surround the bone and attach themselves to extreme positions. However, in insects a series of muscles attach on one side to the exoskeleton and the other to an apodeme[3] that pulls the next segment (Figure 2.3). Resulting tensions on the exoskeleton caused by self-motion and external influences can be measured with specialised organs called campaniform sensilla (CS). Furthermore, joint movement is monitored by complex and specialised organs called chordotonal organs (CO).

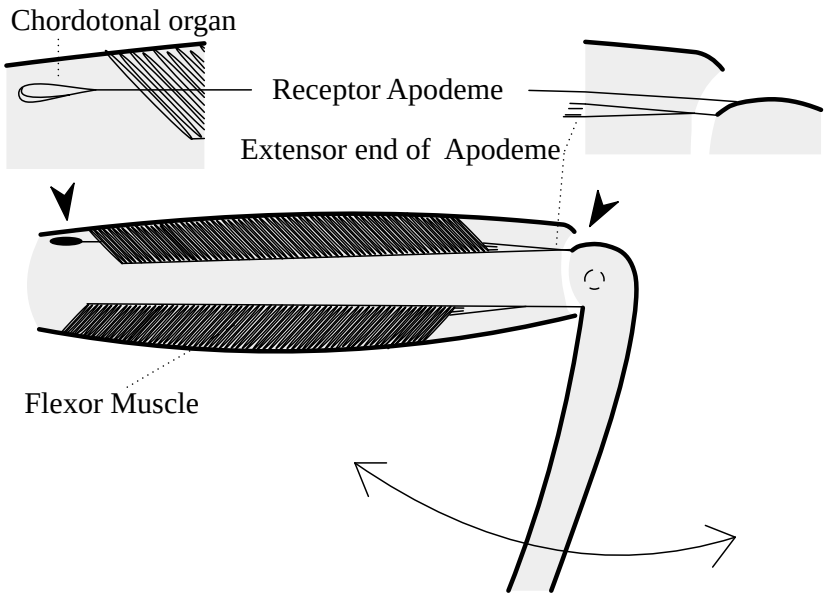


Figure 2.3: Schematic cross-section at the femur

Muscles in insects are located inside the exoskeleton and attached to other segments by apodemes. Segments are moved by pair of antagonist muscle groups; the scheme oversimplifies these, among other things, by showing only one pair of antagonist muscle. The joint position of the femur-tibia joint is monitored by chordotonal organs, which sense position, velocity and acceleration.

[3]Ridge-like ingrowth of the exoskeleton of an arthropod that supports internal organs and provides attachment points for muscles.

Rotation of leg segments is usually accomplished by groups of antagonistic muscles; each pulling in either direction. Muscle groups controlling the BC, CT and FT joints are known as protractor-retractor, levator-depressor and extensor-flexor respectively. Furthermore, muscles pulling to one direction might be stronger than antagonists. For instance, flexor muscles are stronger than extensor muscles (Cruse, 1981; Cruse et al., 2004). By increasing the activity of both antagonist muscle groups, the insect may controls spring properties and different joint tensions (Cruse et al., 1993; Dürr et al., 2003).

2.2.1 Sensors

Depending on where the signal originates, sensors can be categorised as exteroceptors if influences are external, e.g., sensory hairs, hair plates and the campaniform sensilla; interoceptors when sensors are inside the exoskeleton, like the chordotonal organ, strand receptors and multipolar sensillum (Graham, 1985) or proprioceptors (signalling information about self-position or movement (Barnes et al., 1993)). Some insect sensors are categorized as a mixture between these, for instance hair plates are external sensors located close to a joint, whose purpose is monitoring joint position.

Sensors are further divided into tonic and phasic sensors. The former respond to absolute values in the signals whereas the latter only respond to changes in the input. There are also sensors that behave as a combination of tonic and phasic sensors and in most cases levels of adaptation at many stages of the sensor input process can be found (Kittmann, 1997; Wolf et al., 2001).

As mentioned above, insects have a specialised type of sensors for detecting loads, the campaniform sensilla (CS), which detects directional stress on the exoskeleton. Its mechanical properties are no different to stress gauges and similarly CSs can be very sensitive to particular directions. This makes it possible to make adjustments to joint velocities during stance and swing dependent on the load and resistance each leg has (Schmitz, 1993; Graham, 1985). This also includes corrections to anterior and posterior extreme position (Frantsevich and Cruse, 1997; Schmitz, 1993), i.e. transition between stance and swing and vice versa. In addition, the amount of force each of the muscles is required to exert depends on how much weight that leg is supporting.

Chordotonal organs are complex proprioceptors processing various space-temporal information (Filed and Matheson, 1998). The best studied for the stick insect is the femoral chordotonal organ (fCO), monitoring the FT joint (Bässler, 1977; Büschges,

1989, 1994; Sauer and Stein, 1999a; DiCaprio et al., 2002). It has 80 sensory cells on the ventral Scoloparium and about 420 sensory cells on a dorsal Scoloparium, the former being responsible for the feedback control loop in the FT joint (Kittman and Schmitz, 1992; Cruse, 1985a; Stein and Sauer, 1998; Bässler and Stein, 1996). Chordotonal organs are mainly responsible for processing angular position, velocity and acceleration of joints, as well as vibratory signals. The fCO monitors the joint via a receptor apodeme; the fCO is positioned on the back of the femur. This is illustrated in Figure 2.3.

2.2.2 Summary

The stick insect walking system is divided into three thoraces, the pro-, meso- and meta- thorax. Each leg has four main segments, the coxa, femur, tibia and tarsus. All these segments lie on the same plane because two of the joints are parallel hinge joints, the CT (β) and the FT (γ) joints.

Sensors in the leg give information about position, velocity, acceleration and vibration. Furthermore, load can be measured by sensors on the exoskeleton. The model presented in the next chapters only implements signals and sensors described in this section. The CS, the fCO and other specialized sensors (Bässler, 1977) play an important role in establishing movements and coordination between joints in the stick insect's legs. The dependency on peripheral sensory information for patterning leg activity instead of relying on a fixed central pattern generator as might happen with other insects, gives the insect more variety and flexibility of movement required for its natural environment (Bässler, 1988).

2.3 Behaviour

Members of the orthoptera display different strategies for dealing with dangerous situations; for example, locusts perform a long jump that hopefully will separate them from danger; cockroaches have a highly stable and reliable mechanism for running at high speed (Jindrich and Full, 2002); others, like the grasshopper are able to fly large distances. As mentioned above, stick insect's survival skills and daily actions are all related to avoiding being detected by possible predators (Wolf et al., 2001).

This section describes behaviours progressively increasing in complexity. First it is important to know if legs behave independently and if so how they respond to different

situations. The behaviour of the single leg and how it responds to ‘simple’ stimuli is described in section 2.3.1. Nonetheless, the leg’s main purpose is to move the body and for that, legs frequently step. Counterintuitive at first, given the right conditions individual insect legs can step with their own rhythm, and yet, the body does not fall. In section 2.3.2, the mechanisms that have been proposed to allow body stability are presented. These are based on local leg coordination rules applied equally to all legs. On the other hand, pairs of legs are known to behave differently. This is one of the most important features explored in the following chapters and section 2.3.3 presents previous work that supports this idea. As previously mentioned, the control of turning is first investigated, because although forward walking is a complex behaviour, it is prone to exhibit stereotypic movements. Furthermore, turning emphasises the need for thoracic differentiation and brain commands. Section 2.3.4 presents previous research on the study of turning for various animals.

2.3.1 Single Leg

Remaining motionless requires rigidly maintaining joints at a particular position and opposing external disturbances when encountered. The simplest reflexes can be found on individual joints. For instance, to remain motionless stick insects have a high-gain negative feedback controller at joints. These ‘resistant reflexes’ are found when the insect is not walking. However, when the insect is active, there is a mechanism in joints that seems to follow external forces instead of opposing them; this is known as the ‘reversal reflex’ (Bässler, 1976).

It was mentioned above that joints are controlled by many muscles and hence many neurons fulfil the same purpose, i.e., joint torque activation. Therefore, strong coordination between those muscles and the neural networks controlling them are expected to be found. Each muscle population controlling joints has been described as a relaxation oscillator, nevertheless, strongly modulated by sensory feedback (Bässler and Büschges, 1998). Single joint controllers are highly interconnected with others (Dürr et al., 2003; Fisher et al., 2001) and their coordination has been well studied in many circumstances (Frantsevich and Cruse, 1997; Brunn, 1998; Hess and Büschges, 1999). Patterns seen by joints in each leg occur without information from other legs (Fisher et al., 2001) or when these have to behave in a different way (Bässler and Büschges, 1998). Therefore, each leg can behave independently of other legs, although when present, other legs modulate their behaviour (Brunn and Dean, 1994; Dean, 1989).

One of the most complex behaviours each leg performs is the walking step. This is divided into two step phases as mentioned in chapter 1: stance and swing (Cruse, 1985b). The stance, a.k.a power stroke, is when the leg is touching the ground and supporting the weight while moving in coordination with other legs. What is commonly labelled as swing phase for legged animals, has been identified for the stick insect as an interrupted searching for ground (Dürr, 2001). The swing-search phase occurs when the insect moves the leg to a forward position in search of a reliable foothold. The important difference to note is that legs do not expect to find reliable support at any point, i.e., legs are well prepared for not finding ground at the same location as the previous step. This swing-search phase is not clearly apparent on flat surfaces, however, because of rotational variation in the body, legs touch down at different local heights. Consequently, the exact anterior extreme position (AEP) is unknown until load signals indicate leg support is sufficient. Hereafter, the swing-search phase will be simply referred to as the swing phase for simplicity.

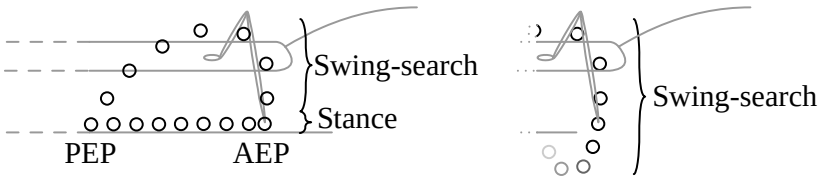


Figure 2.4: Leg step

Leg step phase is divided in two. The stance or stroke is when the leg is in contact with ground, supporting the body and moving it. The swing-search is when the leg protracts to an anterior position. The time between swings is the step period T, which is the protractor time T_p and the retractor time T_r. The swing-search phase is terminated when the leg can support the body.

When the proper conditions are met, an abrupt transition between phases occurs; as shown in Figure 2.4. Swinging the leg is simpler than stance because in the latter, all legs are coupled via the ground and the load on each leg is variable. Consequently, swing can be controlled only by spatial variables[4], whereas stance needs at least to combine position, velocity, acceleration, load and intersegmental information.

[4]Insects also have an obstacle avoidance mechanism when hitting objects while swinging; for this, it is simpler to have stress sensors active while swinging.

2.3.2 Leg Coordination

Just as each joint needs to share information with other joints for achieving proper coordination, the sharing of information between legs is also essential. The stick insect evolved a mechanism for passing information from front to back about where reliable footholds exist. Most of the sensory information is shared with adjacent legs through intersegmental neurons; apparently, fewer than expected (Brunn and Dean, 1994). Influences between contralateral legs are also present; however these are not as strong as the ipsilateral.

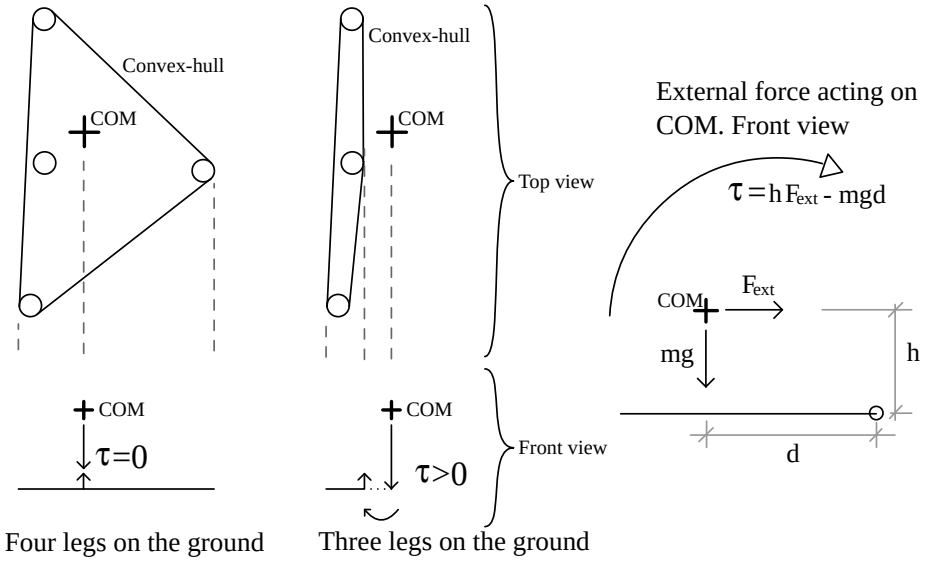


Figure 2.5: Polygon of stability

Static stability is estimated by projecting the centre of mass (COM) to the convex-hull created by legs in contact with the ground. Left: With no external forces and when the COM projects inside the convex-hull, the resulting torque is zero. Middle: On the other hand, a COM projected outside has a resulting torque that could eventually tumble the body. Right: Even if the COM is within the convex-hull, an external force is likely to tumble the body when its projection parallel to the ground is such that the resulting torque is positive, i.e., $\tau > hF_{ext} - mgd$

. Where, h is the height of the COM and d is the distance from the COM to the edge.

Foothold information is first obtained by the antennae[5] and is further combined with the front legs' 'knowledge' of the surface. For instance, if the antenna detects

[5]For a comparison with the cockroach antennae please refer to (Noah et al., 2001) and (Comer et al., 2003).

a gap in front, the insect does not stop until the front legs do not detect an expected contact with surface. On the other hand, if the antennae detect an edge after a gap, the insect keeps trying to find the other end until it succeeds (Bläsing and Cruse, 2004b). Once the front leg finds a reliable foothold, the middle leg tries to step near the zone where the ipsilateral front leg was, and the hind leg does the same with respect to the middle leg. When legs step on the anterior tarsus, the reflex that follows is that of lifting the leg again and placing it in a different place (Schmitz and Haßfeld, 1989). The 'targeting' behaviour appears to be absent only during short steps (Bläsing and Cruse, 2004a) when the velocity of the insect is close to zero[6]. It is important to remember that not only are phase transitions induced by other legs necessary for leg coordination, but each leg also needs to consider its own state. For instance, if legs are supporting much of the body weight they should avoid swinging. Similarly, stance is not initiated until that leg can support the body based on load signals.

Insect gaits are stable most of the time; this can be calculated by projecting the centre of mass[7] (COM) onto the convex-hull[8] created by the tarsus in contact with the ground (Todd, 1985). When the COM projects outside the convex-hull, gravity pulling the COM creates a torque that would cause the system to tumble. This is shown in Figure 2.5. Furthermore, the closer the COM to the edge the more likely it is that an external force F_{ext}, will produce a tumble torque. The condition is met when $F_{ext}\,h > m\,g\,d$; where d is the distance to the convex-hull edge. Note that the convex-hull analysis does not consider adhesive forces.

The most common gaits are the tetrapod gait (in which at least four legs are in contact to the ground) primarily used at low speed and the tripod gait (in which only three are touching the ground at a time) mainly used at higher speeds. These are shown in Figure 2.6. The tripod gait is characterised by consecutive lifting of ipsilateral front and hind legs with the contralateral middle leg. The tetrapod gait is better described by a wave of stance-swing transitions on ipsilateral legs that is not in phase with contralateral legs.

Obtaining common gaits by means of a global central pattern generator (CPG) is straightforward, as it is only necessary to specify bi-stable signals with different phases for each leg. However, because the stick insect walking pattern generator is not centralized, the coordination is achieved by influences between legs that guarantee to some

[6] Under this circumstances many other walking coordination rules are not present (Bläsing and Cruse, 2004a).

[7] In a uniform gravitational field the COM (or centroid) is a point of balance

[8] Minimal convex set containing all points in a set

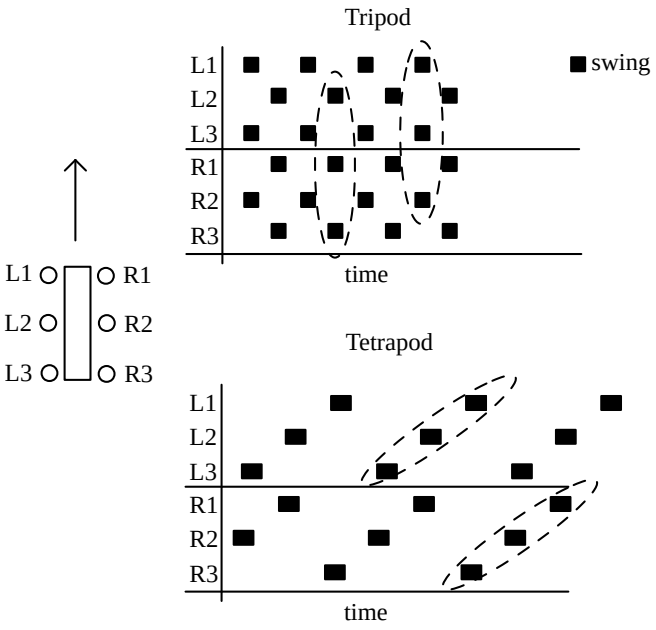


Figure 2.6: Insect gaits

Most common gaits used by insects. Tripod gait is characterised by alternating ipsilateral front and hind legs with contralateral middle legs. In the tetrapod gait swings are loosely coupled contralateraly.

extent the stability of the body (Cruse, 1990). Details of these influences will be discussed later when implemented in section 5.1. This stability and coordination is found even while the insect is walking backward (Graham and Epstein, 1985). Transitions between the tetrapod and tripod gait are more naturally and smoothly generated, and these arise depending on the velocity of the system (Cruse et al., 1991); just as with real insects.

The leg coordination rules' coupling strength is usually assumed to be constant. This approach is known to work even during variation in walking speed. Nonetheless, it has been suggested that rules' coupling strengths for other behaviours need to be adjusted accordingly. For instance, while crossing large gaps walking speed is reduced almost to zero and some rules are not valid if coupling strengths remain fixed (Bläsing and Cruse, 2004a). Additionally, it has been suggested that during turning coupling strength increases on legs on the outside of the turn (Dürr, 2005).

2.3.3 Thoracic Differences

For the stick insect it has been noted that the legs on different thoracic segments have different intrinsic walking directions (Bässler et al., 1985) and can behave differently during specific tasks, for instance, while crossing over large gaps (Bläsing and Cruse, 2004b) or climbing obstacles (Cruse, 1976a). This is also evident for other insects as well, for instance in the cockroach (Jindrich and Full, 1999a; Sathya et al., 2001), and the locust (Bennet-Clark, 1975). Furthermore, the pattern change for each thoracic segment is different depending on the type of surface conditions. For the stick insect, the front legs along with the antennae are more specialised in searching for ground surfaces (Cruse, 1976b; Dürr et al., 2001). For instance, when faced with complicated obstacles, front legs act as feelers and need to be freer from supporting body weight than middle and hind legs. Furthermore, this behaviour is also present while walking on a flat surface, but not frequently seen while walking on continuous thin paths. Perhaps because such paths are commonly found in its natural environment with which the stick insect is more familiar; therefore, there is no need to search for alternative foothold. Foot contact is also different and similarly critical.

When the stick insect needs to walk not only over different texture surfaces but also with different leg geometries (Frantsevich and Cruse, 1997), the range of leg movements for each thoracic segment varies considerably. In this case, adjustments are related to the overall body weight distribution. In general, the hind leg tarsi average position is always behind the BC joint of that leg, whereas for the front legs, the tarsi are in front of the joint. Because the stick insect does not walk with a dynamic stability, the front leg in this arrangement decelerates the body at each step and the hind leg contributes more to the acceleration. The middle leg average position tarsi is similar to the BC joint, therefore, it contributes to both depending how much weight it is supporting and its relative position to the BC joint (Cruse, 1976b). When standing upside down, hind and front legs are no longer in compression, but in tension, so the acceleration contribution of each thoracic segment is the opposite.

2.3.4 Turning

Turning is the change in heading over a period of time, which can be done while running (Jindrich and Full, 1999b), walking (Graham, 1972) or on the spot (Frantsevich and Cruse, 2005). It can be initiated by reactive movements (Camhi and Johnson, 1999), brain commands (Domenici et al., 1998) or a combination of both (Poulet and

Hedwig, 2005). The degree of turning can also be of different magnitudes (Strauß and Heisenberg, 1990; Frantsevich and Cruse, 2005). Consequently, different approaches have been used for the study of turning and hence incompatible results can be found in the literature on turning.

Turning depends on the stimulus that initiates it, whether it is externally (environment) or internally induced (brain intention). For instance, a fast object approaching (external) might cause a fast reactive response, whereas a path integration homing (internal) might induce smoother turns. Stimuli can be continuous (Cruse and Saavedra, 1996), transient (Zolotov et al., 1975) or spontaneous (Strauß and Heisenberg, 1990). Continuous stimuli are useful because it is possible to isolate mechanisms that relate more to turning. Nonetheless, different behaviours could be the same behaviour modulated differently, hence it is important to study transient change (Dürr and Ebeling, 2005). Experiments have been done with tethered or free insects, which normally determines if the control situation is in open or closed loop.

Zolotov et al. (1975) studied bees turning towards a visual stimulus. Interestingly, for the same final heading after the turn, insects produce two different body trajectories, either smooth (slow turn) or sharp (fast turn). This suggests that turning is not a reflex response and that different turns might be decided in the brain. Cockroach walking behaviour differs from the stick insect, suggesting that turning could be accomplished in a different manner. Nonetheless, it has been noted that individual legs in the cockroach behave differently, some of them supporting and other opposing body rotation (Jindrich and Full, 1999b). This will be demonstrated for the stick insect in chapter 3.

Preliminary results (Rosano, 2004) analysing spontaneous turns in the stick insect and concurrent research on leg kinematics by Dürr and Ebeling (2005) showed that legs greatly change stance direction while turning. Furthermore, gaits vary strongly and leg coupling is reportedly adaptable for turns (Dürr, 2005). Experiments by Dürr et. al. consisted of partially fixing the insect's body on top of a sphere with a large-field visual motion stimulus. Insects were constrained to horizontal translations and rotation, i.e., only body height variations were allowed. Results support the fact that visual stimuli play an important role for walking as in (Jander and Volk-Heinrichs, 1970).

It was found that the front legs respond faster and stronger than other legs. In addition, each leg changes step frequency and stance length. One of the most important results is the fact that there is a transition between the steady state curve walking and

straight walking. This, as well as the variability in stepping, demonstrates that joints movement for turning is not pre-programmed, which further supports ideas presented in section 1.5. The tarsus of the hind inner leg (HIL) relative to the body is reportedly arrested relative to the body, however, detailed inspection indicates continuous leg motion. Therefore, it cannot be assumed that the HIL actively holds its position to act as a pivot. Work by Dürr and Ebeling (2005) and preliminary results (Rosano, 2004) show similar single leg trajectories. Nonetheless, reconstructed body trajectories present differences, particularly lateral body drifting seems to be larger in Dürr et. al.'s experiments.

Traditionally, the step frequency and stride length have been used to study turning, and in turn, these parameters have been used for controlling turning. Nonetheless, it is important to note that in all turning situations and for all legged systems, there is always a change in stance direction and speed, which is a direct consequence of the kinematics of the system, i.e., does not depend on what is controlling the legs or the body. These changes in stance direction and velocity can be extreme (Zolotov et al., 1975; Graham, 1985). Subsequently, and depending on the mechanical system, changes in tarsi velocity require a longer or a faster stance phase; or a combination of these (Graham, 1972). Therefore, controlling tarsi direction and speed during stance suggests a more logical approach.

2.3.5 Summary

Joint controllers seem to respond differently if the stick insect is standing or walking. Muscles controlling joints are highly coordinated, however, joints within legs present more flexibility. As a result, legs can walk at their rhythm independently of others. Similarly, pairs of legs behave differently depending on their position along the body. Nonetheless, despite showing high independence, all legs on stance move in such a way that they can achieve complex behaviours, e.g., climbing, turning or crossing gaps.

Stepping is a complex behaviour divided into two phases, stance and swing. The former is terminated by coordination influences from other legs; thus, the position (PEP) where the transition takes place depends on many factors. Swing is terminated by the leg's detection of ground (AEP), which is signalled by load sensory information. There are two resulting gaits favoured by the stick insect, the tripod and the tetrapod.

Particular interest was focused in the turning behaviour because it represents a sufficiently complex motor control and behavioural problem. Particularly the stick in-

sect's slow walking speed while turning, variability of gaits and directionality of legs requires controllability at all levels, i.e., from inter-joint to brain command. Consequently, it is particularly attractive for the study of wider behavioural situations. In chapter 3 the control of turning is analysed in more detail.

2.4 Neurophysiology

In the previous section, it was mentioned that each thorax behaves differently and that joints within legs are not controlled by simple reflexes or a central controller. However, at the beginning of this chapter it was mentioned that mimicking higher-level behaviours with little regard to internal biological control architecture is not the approach to follow. In this section, the elements of known neurophysiology that could be implemented in a robot are explored.

The nervous system of an insect has a series of ganglia interconnected by a ventral nerve cord. The most sophisticated concentration of neurons is located in the head, related to the concentration of various sense organs at the front. It is subdivided in supra- and sub-esophageal. The main sensory processing units in the brain are found in the supraesophageal ganglion, which consists of three parts: the protocerebrum (eyes), deutocerebrum (antennae) and tritocerebrum. The brain acts as an overriding control of reflex and spontaneous lower level behaviours for walking. Decapitated insects are known to be hyperactive or completely stationary, but still capable of walking under certain conditions (Bässler et al., 1985; Bässler, 1988). However, memory and specialised behaviour cannot be attributed solely to lower-level controllers of the nervous system; therefore, commands sent from the brain are of special interest for complex walking behaviour.

The thorax is further divided in three main sections as shown on the right hand side of Figure 2.1. Each section has a ganglion controlling legs on that thoracic segment; these ganglia have a pair of bilateral connectives to other ganglia in the thorax (Dean, 1989). Each thoracic ganglion has its own walking pattern generator capable of controlling walking in a coordinated fashion without other thoracic ganglia involved in the process (Bässler et al., 1985; Bässler and Büschges, 1998). Within each leg, motoneuron activities are influenced by other joints through spiking and non-spiking[9] interneurons, (Hess and Büschges, 1999; Akay et al., 2001; Bucher et al., 2003), i.e., joint activity is synchronised with other joints.

[9]Neuron incapable of producing an action potential

Not all the information is processed within the thoracic ganglion; signals are also filtered in neuropiles[10] of mechanosensors[11], e.g. fCO. Sensory information is mainly processed by spiking neurons, which are more suitable for processing signals with intrinsic temporal properties. Neuron properties like spike frequency adaptation (SFA), restorative depolarising sag or plateau potentials (Schmidt et al., 2001) are highly temporal dependent. Spiking neurons are known to process information about position, velocity, acceleration and vibration on the fCO (Büschges, 1989, 1994; Sauer and Stein, 1999a). Some spiking interneurons can also be found directly innervating motoneuron pools (Brunn and Heuer, 1998). However, the proportion of spiking interneurons, compared to non-spiking, is higher the closer they are to sensory neurons (Büschges, 1989, 1994; Brunn and Heuer, 1998).

Non-spiking neurons have the strongest influence over motoneurons (Büschges, 1990). Most of the biological data available for interneuron morphologies and properties are concerned with non-spiking interneurons (NSI) controlling the extensor motoneurons on the FT joint. Moreover, most of the information available for the FT joint refers to the middle leg[12]. A motoneuron junction potential can be either excitatory (EJP) or inhibitory (IJP), however, only the summed activity is reported obscuring some results. On the other hand, mammal muscles are innervated only by excitatory motoneurons. Clearly, the output for a given behaviour cannot be attributed to just one place, as that output is the summed contribution of the many interneuron populations and some direct sensory influences acting on each muscle (Bässler, 1988; Kittmann, 1997). These parallel pathways are known to either support or inhibit the actual movement (Wolf and Büschges, 1995).

When neurons controlling legs are deafferented and de-efferented, pairs of antagonist motoneuron pools oscillate independently. However, 'spontaneous recurrent patterns' (SRP) do occur and could be identified as leg step transitions (Büsches et al., 1995). However, these are only found for brief transitions and not for the whole stepping[13], suggesting that joint motoneuron pools are less connected to each other at a lower level of synaptic connectivity at the ganglion (Büsches et al., 1995). This agrees with results where leg joints showed no strict coordinated motion (Cruse and Bartling,

[10] Fibrous network of unmyelinated nerve fibers

[11] Sensory neuron activated in response to mechanical pressures or distortions

[12] Only eight excitatory and three inhibitory NSI are well documented (Büschges, 1990; Wolf and Büschges, 1995; Stein and Sauer, 1998; Sauer and Stein, 1999b)

[13] Central Rhythm Generators (CRG) controlling antagonist motoneuron pools for a given joint, are believed to be created by means of alternating hyperpolarizing synaptic input (Büschges, 1998), no rhythm depolarising has been found.

1995), supporting the fact that joints are not controlled by a central pattern generator (CPG), but are coupled by means of sensory information, mechanics and the insect's current state (Cruse and Bartling, 1995; Brunn, 1998) i.e. walking, searching, camouflaging.

There have been reports of coordination between joints, especially between the coxa-trochanter and the femur-tibia. Brunn (1998) reports activity of three non-spiking interneurons innervating motoneurons of those joints while the insect is active or inactive and Brunn and Heuer (1998) were able to classify some reflexes related to the end of swing and modulation of motoneuron pools for those joints during walking. In the same way, Büschges in (Hess and Büschges, 1999) and (Bucher et al., 2003) report influences of the fCO on motoneurons controlling the CT joint while the insect is active and inactive and (Akay et al., 2001) discuss the role of signals from the coxa-trochanter joint affecting motoneuron pools on the FT joint. In all cases, the importance of proprioceptive information and interneuron connectivity is clear for joint coordination. These behaviours or complex reflexes are first divided depending on whether the insect is active or inactive (Brunn, 1998), i.e. if it is moving or camouflaging. For instance, only when the insect is active does the levatory trochanteral (LevTr) excites the extensor (Ext) and the depressor tronchanteal (DprTr) excite the flexor (Flex). When the insect is inactive the FT joint presents a resistant reflex, which is the opposite behaviour to that shown when the insect is active (Bässler, 1976).

Each neuron has intrinsic properties and characteristics that make the whole circuit extremely complex (Büschges, 1998; Delcomyn, 1999). Furthermore, neurophysiological data is limited and inter-connectivity between NSIs and their input from sensory neurons and spiking neurons is not known. Knowing the interneuron connectivity would be useful because it is possible to extract clues about priorities and control organization at different stages of signal processing (Kittmann et al., 1991). However, even if all neuron networks synapses and structures were known, that would not be the end of it. Joint movements are not only defined by neuron network connectivity, synaptic input strength (Stein and Sauer, 1998) or adaptation to repeatability (Kittmann, 1997), but also the way in which muscles are innervated by different motoneurons at different zones along the femur section (Bässler and Stein, 1996). The extensor muscle is innervated by the fast extensor tibiae (FETi), the slow extensor tibiae (SETi), the semi-fast extensor tibiae (sFETi) and the common inhibitory one (CI1) (Bässler et al., 1996). Furthermore, its population along the muscle and its junction potential are important for the response the joint will have to external influences (Bässler et al., 1996). There-

fore, each joint is capable of performing a variety of behaviours with the same neural network.

2.4.1 Relevance

Information about stick insect's nervous system is not sufficiently complete at present to be used more integrally in a robotic system. Nonetheless, it gives useful insights that contribute to a better organizational understanding of the walking controller. More importantly, the more neurophysiologal information is included in the current model, the easier it will be to close the gap between the artificial and the biological system.

It might be possible that the stick insect already possesses a group of different networks for dealing with each behaviour mentioned in section 2.3. However, based on existing research, it is suggested that different behaviours are done with the same neuron network (Bässler and Büschges, 1998; Dürr and Matheson, 2003).

- Walking control architecture:
  - The brain only modulates walking; does not have direct control of muscle groups.
  - The thoracic ganglia receive different brain commands; hence, each behave differently.
  - Legs operate fairly independently of others.
  - Joints within legs are loosely coupled.
- Sensory information:
  - Angular position and velocity; limited acceleration.
  - Force/stress at the exoskeleton.
  - Estimation of tarssus-coxa position.
  - Limited information about neighbouring legs.

In addition, the fact that the same common circuitry is found even in different species suggests the conclusion that a global and highly complex neural network capable of dealing with different situations based only on self-calibration performed for each novel configuration is necessary. This global network should not be based on a hierarchical scheme, since it has already been demonstrated that individual legs, as

well as leg coordination, are based on a decentralized approach, in which command signals from higher levels are less essential.

The most important contribution from neurophysiological and behavioural data is the fact that joints in legs are not pre-programmed to follow strict paths. This allows for a wider variety of navigational paths, just as those needed for moving along branches and that are essential for the environment in which the stick insect lives.

2.5 Biorobotics

Research on legged robots has been carried out for many years, but only recently has there been an effort to implement strategies taken from biological systems (Albiez et al., 2003). However, it has been impossible to implement a robot based solely on biological data, mainly because it is not entirely known how even the most simple walking mechanism works. Consequently, it is still necessary to complement what is available in the neurophysiological and behavioural literature with traditional control engineering. The reasons are understandable; since at the moment engineering and biological systems cannot be the same in terms of materials and mechanics. Particularly, the difference in actuator characteristics is large; biological muscles are far more complex than motors commonly used (Dickinson et al., 2000).

There are some legged robots that, apart from having six legs, have no clear relation to biological systems (Nair et al., 1992; Ota et al., 1998). In addition, there are many robots simply focused on generating common gaits without considering the walking control architecture implemented by insects (Pratihar et al., 2002; Barfoot et al., 2000); just some of these are mentioned, as their relation to this project is not relevant. It is difficult to evaluate and compare performances between current robots because each team aims to achieve or demonstrate different objectives. Here the goal is particularly focused on applying as much biological evidence as possible into a robot.

Most biologically inspired six-legged robots are based on either the stick insect or the cockroach. There are robots of the latter type that have been implemented using a wide variety of techniques. For instance, evolving neural networks for leg control (Fife et al., 2002), using muscle like pneumatic actuators (Nelson et al., 1997), using mathematical internal models for controlling torques in the joints (Nelson and Quinn, 1998), etc. Most robots based on the stick insect differ from cockroach-like robots in a very similar way to what is found when comparing real insects; the cockroach model is preferred when high speed is required (Schmitt and Holmes, 2000).

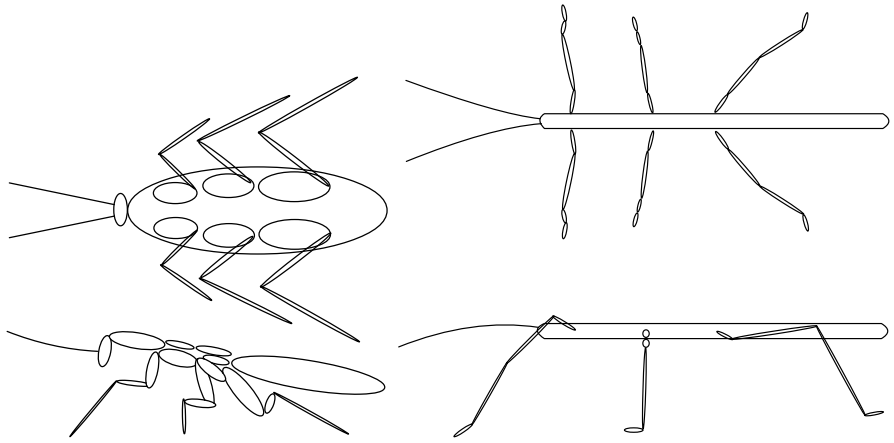


Figure 2.7: Cockroach and stick insect schematics

Left: Cockroach schematic. *Periplaneta Americana*. Approximate size 35-40mm. Stick insect schematic. Right: *Carausius morosus*. Approximate size 72mm.

2.5.1 Cockroach-Based Robots

Cockroach survival strategies are based on running away when detected by a possible predator. Furthermore, cockroaches tend to walk or run in narrow places instead of using camouflage techniques for hiding. Not only does their neuronal architecture allow them to do these things (Pearson and Fourtner, 1975), but also their body anatomy significantly helps (Kingsley et al., 2003; Full et al., 1998). Cockroach leg compactness is evident compared to the stick insect (Figure 2.7), whose environment is mainly open spaces (Delcomyn and Nelson, 2000); this compression helps them to move in small places without further adaptations. In addition, because their body is much wider than it is tall, their centre of mass is very low, so they are highly stable (Jindrich and Full, 1999b; Ting et al., 1994). See Figure 2.5 for a mathematical estimation on stability when $d \gg h$.

Cockroaches depend less on their sensors when moving at high speed, and therefore depend more on pre-programmed central pattern generators (Delcomyn, 1999). Because in these situations leg trajectories are relatively fixed to follow certain patterns, their walking controller is significantly reduced. Robots like RHex (Altendorfer et al., 2001) and the series of 'sprawlettes' (Clark et al., 2001) have implemented this simplified model for solving rapid legged locomotion (Altendorfer et al., 2001). However, robots built for rapid locomotion need special attention because the structure and

dynamics of muscles have a strong influence on the stability of the insect (Full et al., 1998). Furthermore, at those speeds, not only does the geometry of the insect affect the stability of the robot, stability is seriously influenced by state variables (Full et al., 2002)[14].

Apart from the fact that insects walk with certain gaits, other known reflexes have been successfully incorporated into robots. These have been well demonstrated by a series of cockroach-based robots developed by Quinn et al. The first version implemented reflexes such as: searching for a foothold when ground contact is lost; retract and lift reflex when hitting objects while swinging the leg; and joint control stiffness compliance and spring-like properties for joint controllers (Espenschied et al., 1996). The next version implemented a distributed neural network architecture for controlling and coordinating legs (Beer et al., 1992). The first two versions of robots did not have a morphology similar to the cockroach, which as seen in Figure 2.7 is different in each thoracic segment. Furthermore, actuators in robots were DC motors, which behave differently than insect muscles. Therefore, the next series of models and robots were based on a cockroach robot with pneumatic actuators in each joint (Nelson et al., 1997), (Nelson and Quinn, 1999); however, these latter models have only succeeded in standing and not yet in walking.

2.5.2 Stick Insect Inspired Robots

Because stick insects move at low speed and the surface on which they stand is not usually uniform, their walking pattern generator is highly sensory dependent (Bässler and Büschges, 1998). The use of load and stress sensors has been extensively incorporated in robots based on the stick insect. In addition, it has become increasingly common to implement leg coordination rules proposed by Cruse (1990), as they have proved to create stable gaits in a decentralized fashion. Further details of the complete walking model proposed by Cruse et al. are given in section 2.5.3.

Tarry*[15] robots used a combination of techniques for solving locomotion on rough terrain (Guddat and Frik, 2000). The control of the robot is done by using a series of parameters, such as body height, duty factor, step height, etc. These are combined with a kinematical model of the robot in order to propose foot trajectories in cartesian

[14] State variables continuously change, such as velocity and acceleration, whereas parameter variables are relatively fixed for a given system, e.g. inertia and dimensions.

[15] Recently Tarry was acquired by the Biological Cybernetics Group at the University of Bielefeld. Experiments on Tarry belonging to the University of Duisburg-Essen will be marked with (*).

coordinates that are further fed to an inverse kinematical model that specifies the value for all joints. These results are used for training a neural network for each leg in order to produce basic gaits. The control system is then combined with known insect reflexes, such as searching for ground, re-swing and lifting the leg when hitting an obstacle, and so on. Tarry* robots have also implemented a short-term memory map for aiding navigation.

Kimura et al. (1994) proposed a mathematical model based on the stick insect that adapts gaits to external conditions, especially load signals. The walking pattern is self-organized in real time based on load information. The structure of the model was divided according to the physiology of the stick insect. They modelled the interaction that the central pattern generator in each thoracic ganglion has with motoneurons via non-spiking interneurons. Walking was achieved by introducing a local cost function that was optimised according to the energy each leg required. However, the model is focused only on the generation of gaits, and single leg controllers were greatly simplified.

For building a robot it is necessary to have information about the dynamics of the system as well as the control strategies implemented in it (Pfeiffer et al., 1991). Weidemann et al. (1993b) developed a mathematical model of the stick insect's legs for constructing the TUM[16] walking machine. Because the mathematical model is highly dimensional, it was solved for a particular criterion. According to walking insect data, this condition is believed to be related to a minimization of leg bending load coupled with minimization of interaction forces in the walking plane (Weidemann et al., 1993a). Single legs are controlled by specifying trajectories set by leg coordination connections depending on the current step phase (Weidemann et al., 1994) and leg coordination using Cruse's rules between ipsilateral and contralateral legs. A mechanical description of the robot can be found in (Weidemann et al., 1994) and (Pfeiffer et al., 1995).

It has become increasingly common to use neural network architectures for the control of single legs, since these have an inherent ability to learn and adapt (Berns et al., 1994). LAURON is a stick insect robot based on a hierarchical architecture of neural networks able to learn from examples. Training data for the single leg controller is obtained directly from an insect model and then fed into an Elman network [17]. For leg

[16] Technical University of Munich.

[17] Elman networks are two-layer backpropagation networks, with the addition of a feedback connection from the output of the hidden layer to its input.

synchronization, the training data is obtained from Cruse's rule of coordination, and it delivers parameters of phase and velocity to each leg (Berns et al., 1995). In order to solve problems like obstacle avoidance, turns and collisions, a 'reactive-element' was included. This takes higher-level decisions according to the current state of the robot and external sensors, like optical range detectors. LAURON III was purely based on behavioural control (Gassmann et al., 2001). No neural network was used as trajectories were prearranged using an inverse kinematic model to follow specific curves calculated in joint coordinates. In addition, leg coordination was now centrally controlled, assigning phases to each leg in order to supervise global stability.

A robot particularly relevant to this research is Hamlet (Fielding et al., 2001). It was designed for omnidirectional walking and was partially biologically inspired. Fielding and Dunlop (2004) used Cruse's rules for coordination and noted that for non-straight walking adjustments were needed. This is further supported by the current research (section 5.1). Additionally, Hamlet has compliant motion to aid leg on stance synchronization. Unfortunately, the model is centralized and the control implemented is irrelevant for modelling the stick insect. It illustrates, however, that compliance and leg coordination adjustments are needed for non-straight walking.

2.5.3 Walknet

WalkNet is a simulation model based mainly on behavioural results gathered from the stick insect (Cruse et al., 1998). Coordination rules proposed by Cruse et al. in WalkNet for a decentralized control architecture have experimentally proven to succeed with many robots. Furthermore, convergence of these rules to stable gaits has been demonstrated analytically (Calvitti and Beer, 2000) and the suitability of the approach was verified in (Fortuna and Patane, 2002) when individual legs behave differently. According to (Arena et al., 2002), the stability depends on the connections themselves, not on the individual elements. For this experiments joints were fixed to follow certain trajectory and were reduced to only two degrees of freedom. Each element is referring to a single leg controller. In addition, a reinforcement-learning algorithm based on a static stability cost function was implemented for increasing reliability of the model at many starting configurations (Cymbalyuk et al., 1998). A genetic algorithm was used to find a minimum value for the frequency of the loss of static stability (FLSS) cost function. First versions of the model were focused only on leg coordination, and left the single leg controller with a simplified model that switched

between stance and swing depending on simple signals of load, position and velocity (Cruse et al., 1991). However, this model was sufficient to demonstrate generation and sustainability of gaits, as well as the transition between tetrapod and tripod gaits.

The model was later complemented with leg swing and stance controllers based on a neural net architecture, switched depending on threshold values (Cruse et al., 1995). The idea is to emulate a relaxation oscillator model proposed in (Bässler and Büschges, 1998). Trajectories for swing were trained in the network according to biological data (Cruse et al., 1994); once calibrated, it was able to generalize over a large range of positions. Joints are controlled independently for the stance phase (Kindermann et al., 1998). Angles at the BC and the FT joint are controlled by means of a positive feedback controller in which the velocity, and not the position, is fed back into the loop. As a result, no position or velocity has to be computed, as the leg moves reflexively to its own movements. A negative feedback controller controls the angle at the CT joint, otherwise it would move passively towards the ground because of gravity effects. In addition, introducing a negative feedback controlled bias into the BC joint allows the model to make smooth turns.

Of particular relevance to this research is the work done by Schneider et. al. on different velocity control based architectures for decentralised joint control. One key element that will be used herein, is the use of elastic joints as a mechanism for detecting external influences. Elastic joints allow measuring torques by calculating the bending or joint deviation with position sensors instead of using force sensors. Initial work by (Schneider et al., 2005b,a) addressed the problem of controlling a global angular velocity using local positive velocity feedback (LPVF). The spring system proposed by (Schneider et al., 2006) will be implemented for this research because it mimics the elastic properties of animal joints. The main advantage of LPVF is not requiring prior geometrical or mechanical knowledge and no central controller or shared information between joints. However, one of the disadvantages of early versions of LPVF for the control of legs is the difficulty to voluntarily and continuously correct ongoing motion or oppose forces other than those supporting motion.

Height in WalkNet is controlled by assuming that each leg behaves as a spring; therefore, there is no need to introduce a global height controller as this could be specified with the angle at the CT joint. In addition, the compliance of the system allows climbing obstacles within a certain range.

2.5.4 Discussion on Biorobots

The series of robots developed by Quinn et al. present an increasing development in incorporating biological information into robotic systems. The first robots were almost behavioural-based robots and the anatomy with respect to the cockroach was not accurate. Therefore, particular specialized characteristics of the cockroach were missed; for instance, its compact arrangement of legs and morphology for rapid robust locomotion. It was not until subsequent versions that the morphology was based more on the cockroach; however, for these robots the control methodology for legs was not as biologically inspired as those of the first robots and these robots were not capable of walking.

Robots like the TUM walking machine represent the potential of using biological data for simplifying tasks with many degrees of freedom. Complicated tasks, like leg coordination, are significantly solved in a decentralized fashion by using Cruse's coordination rules. However, leg trajectories were relatively fixed to follow certain shapes taken from real insects. These are sufficient for walking horizontally for a selection of obstacles; however, as argued in chapter 1, performance is diminished because unknown difficulties have to be solved by limiting motion to that similar of wheels. Furthermore, as seen in section 2.4, biological experiments do not support a position trajectory controller for legs (Cruse, 1981), which might cause problems in the model if it is to be complemented with further biological strategies for the single leg control, as both systems are implemented with a completely different approach.

The flexibility of leg trajectories and learning capabilities is important to incorporate in a robot because these need less explicit commands to operate. First versions of LAURON demonstrate these ideas by using Elman neural networks. However, learning input was only related to joint coordinates taken from an insect model. The disadvantage of this procedure is that the robot is mimicking only the final output of the biological neural model, which is not based solely on position trajectories, or joint coordinates. In addition, leg coordination is taught to generate tripod gait, which as discussed earlier, does not work for every situation. Although LAURON III is a good model for successfully implementing insect behavioural reactions, its controller implementation is far too different from the biological one, especially for the leg coordination that is highly centralized. This problem is more evident in robots like the Tarry* robots, which have little relation to biological systems.

The WalkNet model closely resembles the walking architecture of the stick insect.

However, the model as it stands does not work in a real robot implementation, i.e., it is a kinematic model. Furthermore, turning is restricted to smooth curves, and for smaller curvature radii, the simulation does not match insect's leg kinematics.

2.6 Discussion

Not surprisingly, there is a lot of information known about the stick insect. Previous sections are just an attempt to highlight the most salient features of particular interest to include in the model presented herein. However, the more significant were selected with the intention that the rest of the information can be included later on. Each of these will be discussed in turn.

2.6.1 Spiking and non-spiking neurons

In (Rosano, 2004), preliminary results presented a spiking network-based controller for the CT joint, however, parameterization and inclusion of other joints proved difficult and inconclusive. It was clear that the knowledge of neuroscience has to improve further before any real attempt to control intricate dynamic systems with spiking neurons is possible. Moreover, neurophysiological data on the stick insect is biased towards a particular joint (FT) and towards a particular leg (middle leg). Nonetheless, some studies suggest that concentration of spiking neurons are closer to the sensory neurons and non-spiking neurons are close to the motoneurons.

It is important to consider that interneurons closer to mechanosensors are more susceptible to external noise. Furthermore, in order to detect injured sensors or interneurons it is best to have information converging from different sensory neurons and interneurons. Therefore, it is possible to hypothesise that spiking neurons mainly process raw sensory information and that information is then process in the ganglia for output to motoneurons through non-spiking neurons. If so, it is then possible to breakdown the network and concentrate on the information processing at the ganglion and subsequent interneurons controlling motoneurons. This would require mainly the use of non-spiking interneurons that are more similar to artificial neural networks (ANN). Therefore, the simulation model will use the type of sensors available in the nervous system for the stick insect, however, the sensory pre-processing would be omitted.

2.6.2 Independent and compliant

In section 2.4 it was mentioned that joints are not centrally controlled, nor they are controlled by CPGs. They also respond differently depending on internal and external contexts. Moreover, each leg can walk independently and it is just modulated by others when mechanically coupled. This data suggests that legs have their own walking pattern generator independent of other legs and do not require brain commands for basic movements.

On the other hand, joints move in synchrony and in turn, leg movements are coherent with the direction the body follows. Similarly, each thorax is known to behave differently for various locomotion tasks, however, each pair of legs collaborates with others. Consequently, whereas most of the low-level (neurophysiology) information indicates legs move independently, most of the higher-level (behavioural) indicates high collaboration.

2.6.3 Solution?

The problem is that legs are mechanically coupled through the ground and one cannot move without needing the others to move accordingly. For a hexapod robot this would require eighteen joints centrally controlled. One proposed solution by (Schmitz et al., 2001) is based on the ‘reflex reversal’ (Bässler, 1976) and its implication in control theory. The model is a set of non-stiff joints whose deviation between the set angle (set-point) and the real angle (measured) is under positive feedback control. If one joint is moved, the others need not calculate the position that would solve for the new posture, it is only necessary to ‘follow’ passively, thus solving the closed chain kinematic problem.

However, there are some concerns about implementing the controller as it is. The deviation between the set-point and the real angle is caused by a resulting torque, nonetheless the whole of the system is interconnected. How can one joint discern torques that produce walking, climbing or turning and torques produced by external or internal forces that impede walking? It is known that legs always respond negatively to external disturbances (Bartling and Schmitz, 2000). However, this mechanism has been assumed to emerge only when external forces are applied. Therefore, these controllers assume that leg controllers can differentiate between internal and external forces based on the magnitude of the force felt; responding to internal or small forces with a positive feedback and external or large forces with a negative feedback.

Control versions based on local positive velocity feedback (LPVF) prior or concurrent to those presented in chapter 4 focused its tests on crank turning (Schneider et al., 2005b,a). The mechanics of the systems, a two degrees of freedom limb, were not sufficiently complex to be transferred directly to a hexapod robot. Local joint controllers worked by sustaining current velocity and moving in the direction of the bending. The difficulty is that these sensed variables, velocity and bending, are not always in agreement with how joints should be moving if they were controlled centrally. Additionally, the control architecture allows for little voluntary control intervention of leg trajectory.

2.7 Problem and Approach

The target system, the stick insect, has been presented, as well as the prospective problems to tackle. This section now turns to the discussion in more detail on what is to come in the next chapters.

The constraint for the stick insect model is that no single neural concentration controls all neurons, neither in the brain nor in thoracic ganglia, i.e, the model needs to be decentralised. Joints, legs and pairs of legs are independent at various control levels, but all collaborate synchronously when required. If one solves the problem for the individual joint there is no guarantee that it would work when interconnected with the rest of the system because, as said before, the nervous system response is context dependent.

Current proposals lack successful combination of two known aspects: leg compliance and leg's own intended motion. The positive and the negative feedback mechanisms are known to be present in the stick insect but their integration into one system varies. The negative feedback has been previously relegated to an emergent reflex because it has only been shown when the leg is perturbed. However, if is not possible to detect a negative feedback without perturbing the controller, what other clues can be used to infer a continuous negative feedback? Moreover, legs ultimately need to be controlled or directed by the brain. If the leg intention is under negative feedback, then it is more likely that this first intention is based on brain commands because the positive feedback can be solved locally. If so, it would be necessary to investigate how simple the command needs to be and how it is processed by different legs.

Work by (Schneider et al., 2005c) suggest that in order for joints to produce power, the joint cannot depend solely on positive feedback. This model works by switching off the positive feedback depending on the mechanical power generated by the joint. A

controller implemented in (Rosano and Webb, 2006) and concurrent work by (Schneider et al., 2006) improve this controller by replacing the discrete behaviour with a continuous response. Different approaches differ in the level of decentralisation and most importantly in the reference used for setting angular speed. Not being able to specify a tarsus trajectory with the accuracy observed on behavioural experiments led us to search for alternative options other than those based on LPVF.

Particular interest is deposited in experiments that require all control levels, the brain included, that together solve a particular task. Furthermore, the body and leg kinematics will be analysed in detail and thus, non-invasive experiments will be favoured. The experiment chosen is the kinematic study of body and legs of the free stick insect guided to turn towards a visual target. This experiment shows a direct link between visual input and body direction, hence, leg kinematics. The variability of leg movements is an ideal opportunity to propose a model where the use of any CPG is impractical. The experiment setting and analysis is presented in chapter 3.

Hypotheses are then implemented in a dynamic simulation model that resembles that of WalkNet. However, leg co-ordination rules needed to be adjusted to cope with turning and a GA algorithm is used for parameter calibration. Hypotheses drawn from experiments are incorporated in the robot simulation. These are the ability to calculate the position of the target and the fact that legs continuously control their own intended motion. The control of the single leg, its directionality and its compliance with other legs needs particular attention. The proposed single leg controller is described in chapter 4.

This thesis demonstrates that the proposed single leg controller fulfils requirements of directionality (brain command) and compliance. However, the control of turning by the stick insect requires analysis of the particular role of the legs on different thoracic segments. It is also demonstrated that, not only the intended direction needs to be controlled, it is also necessary to balance leg contributions in each thoracic segment. This thoracic differentiation for the control of turning is presented in chapter 5.

Biological implementation of this model is further supported by implementing the single leg controller using an ANN for its control. The approach followed and the resulting network topology is described in chapter 6.

Chapter 3

Behavioural Experiments on Stick Insect Turning

This chapter presents the behavioural study of stick insect locomotion. It starts with an overview of turning analysis techniques (section 3.2) which, perhaps for historical reasons, is biased towards wheeled vehicle analyses. As mentioned in section 2.7, non-invasive experiments with freely walking insects will be preferred. Particular attention will be paid to detailed motion of legs and body, hence high quality images of the walking sequence were required. It was decided to program a tracking algorithm that would compensate motion of a handheld camera moving freely above the insect. The closeness of the camera to the insect compensates for the lack of video resolution and avoids further constructions for camera support. This program has various analysis tools programmed specifically for the study of insect locomotion; details are explained in section 3.3.

The experiment setting is described in section 3.4. It describes the arena and the visual stimulus that controlled the insect direction. In section 3.4.1 the methodology used for the study of turns is presented. Furthermore, two different behavioural experiments were needed to discern between thoracic contributions for turning. First, experiments are shown with intact stick insects in section 3.5. However, the front leg's major role for turning obscured contributions from other thoracic segments. In fact, this large contribution by front legs led us to initially relegate middle and hind legs to have just a passive role for turning (Rosano and Webb, 2006). However, discrepancies in leg trajectories between the real insect and this model led us to propose a second behavioural experiment that highlights middle and hind leg contributions. In section 3.6 a similar experiment is carried out with the front tarsi blocked. Analysis of body

and leg motion relative to the visual target is in section 3.7; conclusions are presented in section 3.8.

N.B. Some results in this chapter have been previously published in (Rosano and Webb, 2006, 2007).

3.1 Review of Turning

Stick insect's specialisation in camouflage proves to be advantageous for the study of freely orientated curved walking. If they "realise" their stealth-cover is no good, they will continuously try to find a place to hide. However, their escape is not directed towards a random direction. According to Jander and Volk-Heinrichs (1970), insects are strongly attracted towards objects with darker tones and with shapes similar to branches. Therefore, it is possible to stimulate them until they start looking for a better hiding place; then one can direct their walk by introducing an attractive visual stimulus. Smell might play a role in directing the behaviour, however, the speed at which the object was moved and the reaction time of the insect do not seem to support this sensory information.

In section 2.3.4 recent research on stick insect turning behaviour was presented. It was demonstrated that front legs are the first to change direction given a visual stimulus. Additionally, the middle inner leg also shows faster changes some time after the front legs. However, the changes that legs undergo during turning are of particular interest. Some of the most salient questions that motivate this research are presented below:

1. What and how complex is information sent from the brain?
2. How do legs process information from the brain?
3. Do middle and hind legs respond passively or actively?
4. Are changes in leg coordination part of the turning mechanism or a secondary effect?
5. What commands control leg direction and stepping?

These questions are particularly related to the control of turning of the stick insect. However, as mentioned in section 2.7, the last (5) is an open question for straight

walking as well. As mentioned in section 1.4, first it is necessary to study the elements legs need for turning, which is the more generic situation. Subsequently, commands can be simplified to verify if the desired straight walking behaviour can be obtained. Note that to answer question 3, it is better to have a situation where front legs do not influence results or have little contribution in the experiment. Their contribution to turning was 'cancelled' by blocking the front tars and the change in strategy by the stick insect was significantly reduced. This is explained in more detail in section 3.6.

3.2 Parameters of Turning

In section 2.3.4 it was mentioned that turning has been studied in various insects and in various behavioural contexts. However, methodologies for the study of turning vary, and therefore results cannot be easily compared. Some experiments focus only on turning 'tendencies' (Poulet and Hedwig, 2005) or leg coordination (Graham, 1972), which do not require further detail. However, a more comprehensive approach should integrate leg kinematics, body motion and stimulus.

Wheeled vehicles only control tangential velocity and there is no body movement relative to the wheel ground contact (chapter 1). The centre of turning, or axis of rotation (AOR), is very well defined, positioned perpendicular to the differential drive. The closer this AOR is to the centre of the robot, the tighter the turn. The curvature is either determined by different contralateral wheel velocity (two wheels); by frontal steering wheel angle (three wheel); or by the tangential projection in Ackermann[1] driven vehicles (four wheels, fig. 1.2).

In contrast, the body in legged systems is not constrained in any degree of freedom and therefore a different approach must be followed. First, the centre of turning is unconstrained and dependent on leg kinematics. In addition, body angle (heading) and walking speed are not directly related. On the ground plane, the body is described by three variables, two for position and one for orientation. However, the reference position along the body is arbitrary because the axis of rotation does not depend on the body geometry alone. It is common to choose the centre of mass (COM) because some dynamic properties relate to it. In addition, it is also possible from that position to calculate the rest of the body. Nonetheless, particularly for symmetric elongated bodies the extreme points along the body convey information more directly and better describe trajectories.

[1] Geometric arrangement in the steering of cars to produce different curvatures in each wheel

In section 2.4.1 it was mentioned that each pair of legs behaves independently. Therefore, it would be possible for each segment to follow different trajectories even though they are mechanically coupled. Furthermore, because walking is not centrally controlled, the kinematics of each pair of legs relates to its own ganglion (thoracic segment). In other words, the relationship between tarsi and their ganglion is stronger than to the COM. Therefore, it is better to study the body in the same way that appendages, compound-pendulums and other similar objects are tracked; at their extremes. This differentiation will be demonstrated more clearly in section 3.5.

The radius of curvature is usually used to describe how tight a curved trajectory is. As mentioned earlier, it is straightforward to calculate the axis of rotation for wheeled vehicles. However, the AOR could be anywhere for legged systems, furthermore, it continuously varies with time. Biomechanical analyses usually report the instant axis of rotation because animal articulations are composed of joints that do not have a fixed pivot. Angular motion in such joints is achieved by contact between curved surfaces that are not circular, thus resulting in an angle-dependent instant axis of rotation (IAR). The approach for applying this technique to the study of body motion is explained in section 3.3.3. Additionally, it shall be demonstrated that the IAR contains the same information as the curvature of the body trajectory, however, the IAR is more precise about the location.

3.3 Tracking Algorithm Overview

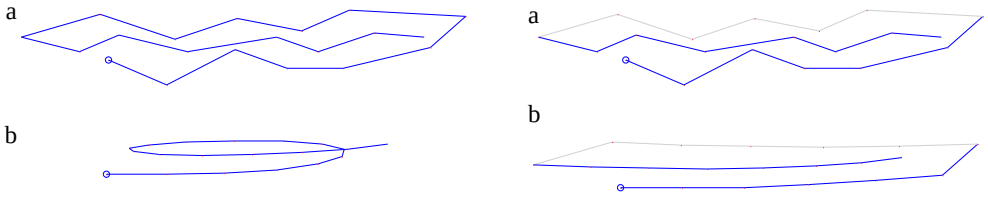
Before going into more detail about the experimental setting and results, it is important to describe the algorithm used to study the stick insect.

The tracking software operates based on a colour classification scheme that allows the user to specify colours by sampling images; this is explained in more detail in appendix A.2. The stick insect was painted with three marks in the body and in each tarsi. Colours are specified as marks and found throughout the sequence of images. Each tracked region is grouped as a single sequence of marks. For details on the graphical user interface and alternative options not mentioned in this section, please refer to appendix A.3.

Mark positions were given by the average position of the largest group of pixels belonging to a given compound colour. This is a relevant feature because some marks were composed of 10-20 pixels, therefore, even motionless objects have some variability on the position. Once all marks were identified in all frames, these were manually

labelled. Labels were: *Body*, *Stance*, *Swing* or *Reference*. Therefore, each mark in a sequence was characterised by its position $\vec{M}_k$, label and timestamp, the colour information was not used for further processing.

Once all mark-sequences are properly labelled there are two options for transforming camera motion. All marks can be tracked relative to the body, which shows individual leg trajectories; alternatively, marks can be shown relative to tarsi on stance, which shows body motion on the surface. The latter transformation can be applied considering either camera rotation and translation or just translation. This transformation, either including rotation or not, is the basis for extracting all parameters describing turning.



(a) Continuous smoothing (b) Step-phase divided smoothing

Figure 3.1: Smoothing procedure

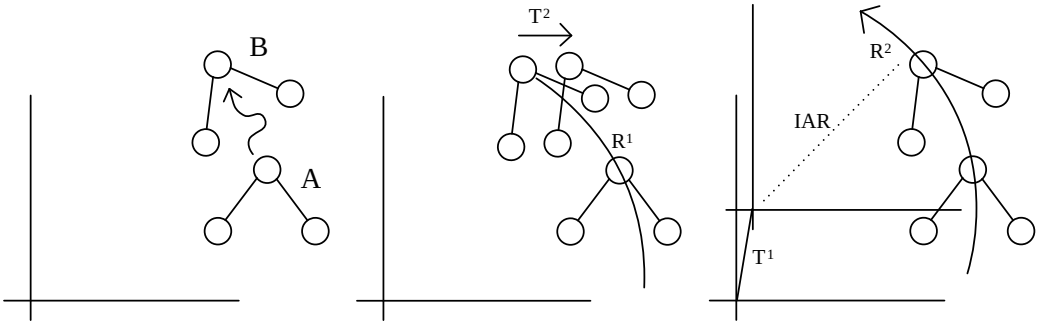
Leg trajectories relative to the body are shown to demonstrate the different smoothing options. Trajectory (a) is the original path without smoothing and (b) is the smoothed version. Left: Smoothing is applied continuously along the whole mark-sequence. Right: Smoothing is done separately for swing (light colour) and stance (dark colour).

Once camera compensation has been done, the rest of the options are mainly for output and visual aids. Nonetheless, it is possible to select an intermediate smoothing algorithm on mark-sequences representing tarsi trajectories. If the mark-sequence is smoothed continuously trajectories shorten and do not represent the original accurately enough. This is shown in Figure 3.1(a). On the other hand, if swing and stance positions are smoothed independently, leg trajectories still resemble the original. This is shown in Figure 3.1(b). The parameters specified are the amount of marks used for averaging, or mask size, and the number of times smoothing is applied on the same section. The smoothing shown in Figure 3.1 was exaggerated to show the effects of smoothing.

3.3.1 Camera Movement Compensation

Rotation could not be avoided on experiments when the handheld camera was constantly moving whilst following the insect; also vertical movements were unavoidable. To estimate the motion of the camera, it is necessary to correlate invariant features between image frames. It is possible to use a pattern on the background or set fixed reference points on the background. A transformation matrix $H(\vec{\Upsilon}, \theta)_k$ and scale factor σ_k for each frame k have to be found so that the reference point motion between adjacent frames is minimized. Where $\vec{\Upsilon}$ is a translation vector and θ is the angle of the rotation matrix R. If one assumes that those points are in fact the same, the transformation matrix would then describe the motion done by the camera. Note that reference points need not be the same for the whole of sequence, it is just necessary to have reference points between frames.

Unfortunately, there are infinite possibilities for a group of reference points going from A to B, even when the distance travelled is small (Figure 3.2(a)). Considering just the simplest smooth solutions, camera compensation can be done in two different ways, each with different results when accumulated.



(a) Transformation problem (b) rotation on COM (c) rotation on IAR

Figure 3.2: Defining Instant Axis of Rotation (IAR)

Left: There are an infinite number of options to go from A to B. Center: First rotation and then translation. It is numerically more stable. Right: First translation and then rotation. This gives the instant axis of rotation.

One of the simplest approaches is to first rotate the set of vectors describing the spatial distribution of the point, to the same orientation as the next frame, and then translate them to match the next position fully, i.e. rotate by R_k (θ_k) and then translate

$\vec{\Upsilon}_k$; this is shown is Figure 3.2(b). Alternatively, it is possible to find a frame of reference such that movement is only by rotation about (0, 0), i.e. translate the reference axes and then rotate, Figure 3.2(c). Mathematically, both require translation and rotation, and both could be simplified down to a single transformation matrix. In reality however, translation of the axis is just a reference convention. Therefore, at each frame it is possible to calculate a rotation around an instant axis of rotation (IAR) to go from 'A' to 'B' in a single movement.

The first transformation is mathematically less prone to error accumulation because it is not affected by small angles. Additionally, the transformation using the IAR is numerically unstable because if the angle is zero the distance to the IAR is infinite. Unfortunately, due to time constraints and particularly code recycling, the first alternative was not used directly. Instead, for small angles an alternative equation based only on translation was used; this is described overleaf. For angles above a given threshold, transformation H_k is defined as follows.

$$H(\vec{\Upsilon},\theta)_k = R_k(\theta_k)\vec{\Upsilon}_k = \begin{bmatrix} \cos(\theta_k) & -\sin(\theta_k) & \cos(\theta_k)\Upsilon_x - \sin(\theta_k)\Upsilon_y \\ \sin(\theta_k) & \cos(\theta_k) & \sin(\theta_k)\Upsilon_x + \cos(\theta_k)\Upsilon_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

The transformation matrix was then applied to all frames on the sequence.

$$\sum_{k=1}^{K} \left(\vec{\Upsilon}_k \vec{M}_k - \sigma_k H(\vec{\Upsilon},\theta)_{k-1} \vec{M}_{k-1} \right) \approx [0,0,0]^T \tag{3.2}$$

where $k = \{1 \ldots K\}$, K being the total number of frames, and $\vec{\Upsilon} = [\Upsilon_x, \Upsilon_y]^T$. Transformation H_{k-1} was applied to all marks on that frame and the process was then repeated for each time-step. Note that marks at frame $\vec{M}_k$ are also transformed by $\vec{\Upsilon}_k$, therefore, motion with respect to $[\Upsilon_x, \Upsilon_y]^T$ is only rotational. Note that to know the position of the last frame it is necessary to accumulate all transformations, hence the error accumulation.

To calculate $H(\vec{\Upsilon},\theta)_k$, first the angle that minimises the error is found (equation 3.3); then the translation $\vec{\Upsilon}_k$ of the reference is calculated (equation 3.4) and then marks are rotated around that position by θ_k. Finally, the scale is calculated (equation 3.5). Details can be found in appendix A.1. At least three marks are needed to compute H_k. Because insects walked slowly there are usually more marks per frame and estimations improved.

$$\theta_k = \arctan\left(\frac{\sum_{i=1}^{N}\vec{M}_{k,i}\times\vec{M}_{k-1,i}}{\sum_{i=1}^{N}\vec{M}_{k,i}\cdot\vec{M}_{k-1,i}}\right) \tag{3.3}$$

$$\vec{\Upsilon}_k = (R_k-1)^{-1}\sum_{i=1}^{N}(R_k\vec{M}_{k-1,i}-\vec{M}_{k,i}) \tag{3.4}$$

$$\sigma_k = \frac{\sum_{i=1}^{N}\vec{M}_{k,i}\times\vec{M}_{k-1,i}}{\sum_{i=1}^{N}\vec{M}_{k,i}\cdot\vec{M}_{k-1,i}} \tag{3.5}$$

The procedure described above can use any reference mark between frames, where ($\times$) denotes cross product and ($\cdot$) is the dot product. The floor of the arena had no pattern to use as a reference, however, the tarsi of the insect grasp the surface sufficiently strongly to consider the stance phase as a fixed position on the floor. Therefore, marks representing tarsi on stance were considered as reference points, as these hold sufficiently firmly to the ground. Figure 3.3(a) shows data of all legs without processing; no body is shown, stance is shown darker (blue) and swing light grey. Note that because the insect must be kept in the camera's view field there is no identifiable reference. In Figure 3.3(b) each time-step has been transformed accordingly. Stance trajectories are now all projected to the same point, and swing trajectories move relative to them.

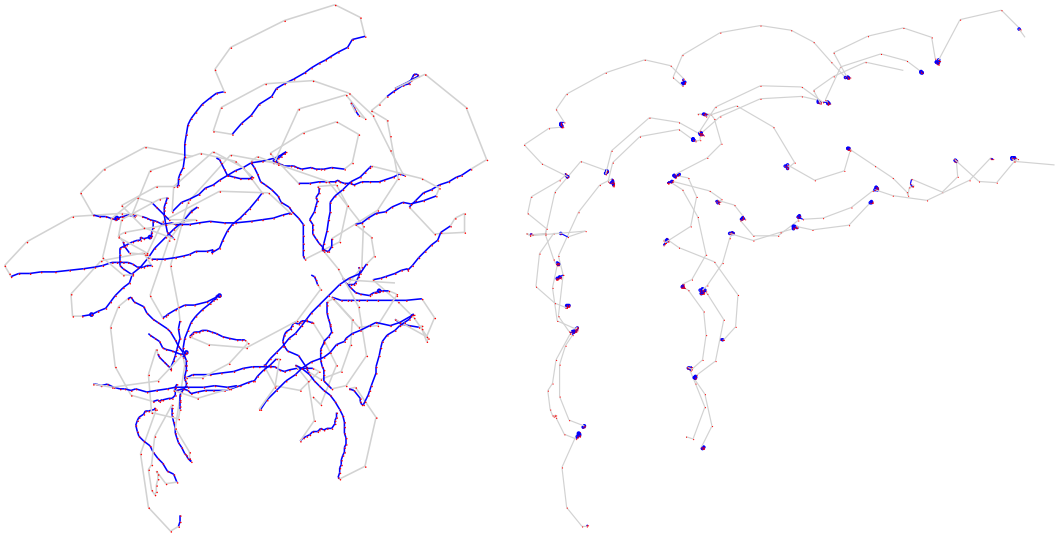
As previously mentioned, when rotation gets close to zero equation 3.4 becomes numerically unstable because of the matrix inverse. When rotation was very small, only translational motion was assumed and the following equation was used for such cases, which matches centre of masses between frames.

$$\vec{\Upsilon}_k = \frac{1}{N}\left(\sum_{i=1}^{N}\vec{M}_k - \sum_{i=1}^{N}\vec{M}_{k-1}\right) \tag{3.6}$$

3.3.2 Cumulative plot of the IAR.

One of the most important analysis tools calculates the instant axis of rotation for the body. The IAR is particularly useful to describe turns because it describes motion by means of rotations. Figure 3.4(a) shows with squares the IAR of each frame, the size of the square represents by how much the body rotated between adjacent frames. It is also possible to display the IAR from frame k and t time-steps before, IAR (k, $k-t$). The IAR is calculated with the same algorithm as that for the camera, i.e. IAR $\equiv \vec{\Upsilon}_k$ given by equation 3.4. However, now it is the points on the body instead of tarsi on stance that was used as input vectors in the equation.

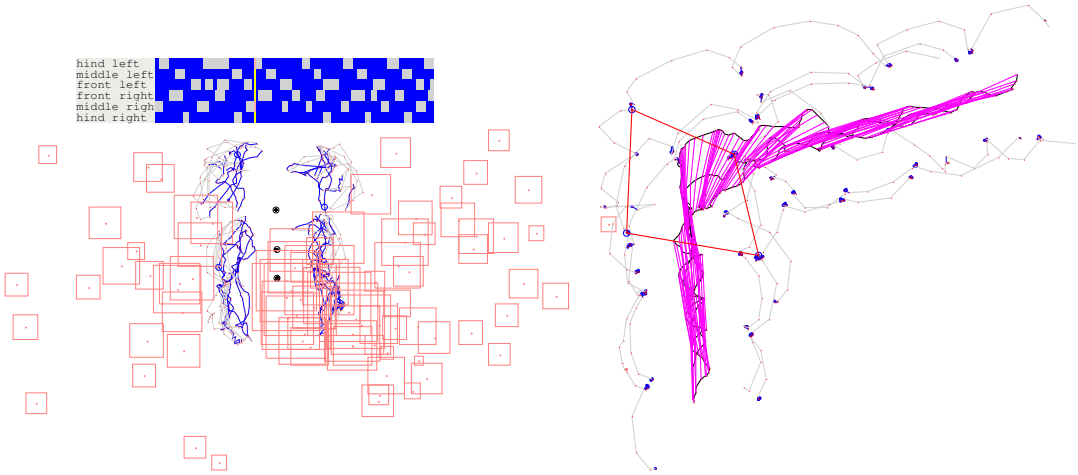
It was mentioned that it is possible to find an IAR for almost all body translations, however, a distant IAR indicates that the motion was mostly translational. Such



(a) Raw data (b) Frames transformed

Figure 3.3: Tracker: Example of tarsus trajectories processed

Left: Raw data of six legs, blue lines (dark) are legs when in stance; grey lines legs when in swing. Right: Trajectories are reconstructed by projecting back adjacent stance marks to the same position.



(a) Holding body (b) Body sequence

Figure 3.4: Tracker: Visual output options

Left: Using the body as reference it is possible to analyse individual leg trajectories. Squares show the IAR, the size of the squares represent the angle of rotation. Right: Trajectories of body and swing marks are plotted with respect to tarsi on stance. Polygon of stability is shown for the current legs on stance, i.e., for one position.

motions usually have a very small rotation angle, because the distance travelled is proportional to the distance to the IAR. Therefore, only rotations larger than a certain threshold were considered as rotational, usually one degree. This error is accumulative and should be considered for long sequences.

The IAR is not fixed to a particular position relative to the body, but for most insect turns its average position is maintained within a certain region depending on the type of turn. To analyse the IAR, the area around the insect was divided with a regular grid and the rotation angles of IARs within each grid was accumulated. The size of the grid was normally set to a quarter of the distance from prothorax to metathorax.

3.3.3 Additional Tools

The tracking algorithm calculates the position of the insect's tarsi and body position. This information can be used in various ways to understand the dynamics of walking. For instance, leg motion relative to the body can be analysed if body marks are processed with equation 3.2 in the same way as stance marks for camera movement compensation; this is shown in Figure 3.4(a).

Further analysis options include displaying the stability polygon of tarsi on stance, displaying a line from metathorax to prothorax, showing the instant direction of individual legs, and displaying the instant axis of rotation. These options can be displayed either relative to the body or to tarsi on stance.

Various functions were programmed specifically for analysing insect locomotion. The program can plot step phases for each leg, showing the walking gait; this is used in chapter 4. Figure 3.4(b) shows a complete reconstructed insect trajectory, tarsi positions are as shown in Figure 3.3(b); the body is represented by a line from metathorax to prothorax. The convex-hull formed by the legs touching the ground is used along with the centre of mass to estimate the torque of the system, i.e. it estimates how likely it is to tumble to one side. This stability polygon is shown at a given time in Figure 3.4(b).

N.B. If this thesis is accessed digitally, it is possible to zoom in on images indefinitely without loss of quality because real coordinates are contained in the file, i.e., figures are vector graphics.

3.4 Experimental Setting

As mentioned in section 2.7, non-invasive techniques for walking analysis which still permitted control of the insect's motion were favoured. The stimulus used is similar to that presented for the study of bees in (Zolotov et al., 1975), where the insect was motivated to walk towards a visual target positioned at different angles. However, the kinematics analysis bears more resemblance to that done by Domenici et al. (1998) for the study of crayfish, which paid particular attention to individual leg trajectories and their contribution to body motion. Results presented in (Jindrich and Full, 1999b) and (Mu and Ritzmann, 2005) are particularly interesting because they demonstrate that individual legs play a major role in curve walking. Leg contribution, and hence turning, varies for different stimuli and with the insect's internal motivation. However, the stimuli (antennae contact) are not easy to quantify (Camhi and Johnson, 1999).

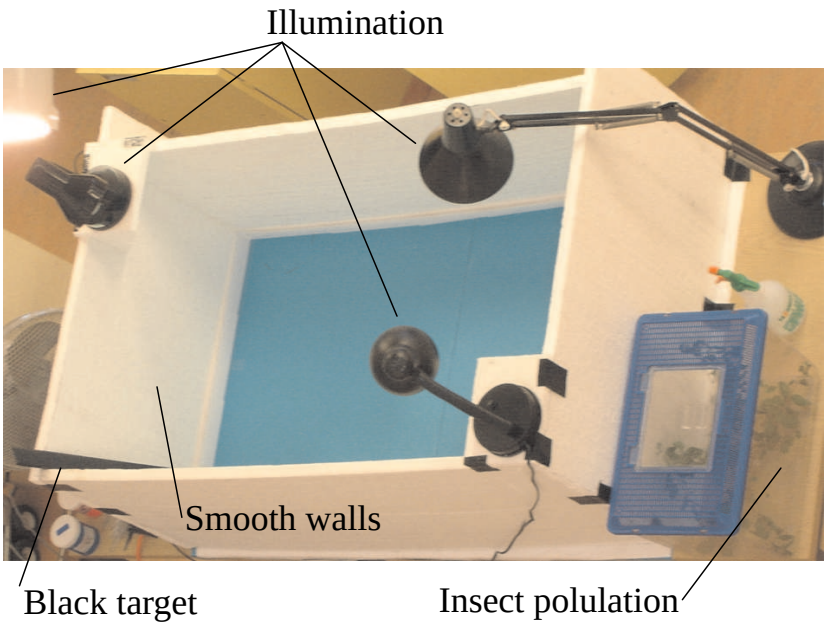


Figure 3.5: Aerial view of the experiment setting

Three lamps were used at the positions shown; the walls were white and were sealed at the lowest point to avoid visual contrast.

Following the idea that solving for a generic situation might lead to a more generic understanding, the visual stimuli for this experiment were positioned at arbitrary angles. The stimulus consisted of a black bar, 4.5 cm wide and 60 cm tall. Adult stick

insects reared at our institute were placed in an arena (67cm by 177cm) with white walls (50 cm tall) around it to eliminate additional visual stimuli. This is shown in Figure 3.5. Insects were first allowed to walk continuously for about one minute before the visual target was introduced. The target was placed within the insect's visual field in a different direction to its current heading, and no more than 30cm away. The animal would reliably respond by turning to walk in this direction. Just before the insect reached it, the target was quickly removed vertically and then repositioned, no more than 30 cm away, inducing another turn. This could be repeated around 10 times before the insect changed its attention to the walls or ceased walking.

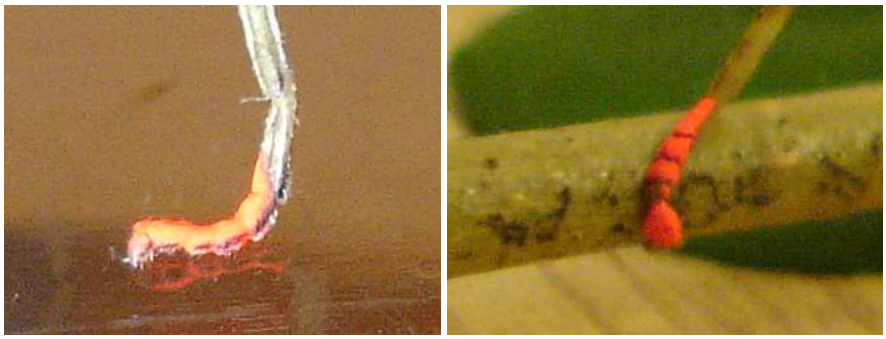
Nonetheless, insects were sometimes attracted to the walls of the arena despite these being white and smooth. The analysis did not include sequences where insects hesitated between the black bar and walls, or when the response caused by the black bar was clearly weak. The first responses to presentation of the object were also eliminated, as these were less consistent.

In the second experiment, the front tarsi were temporarily blocked with dried water based paint[2], so that the role the middle and hind legs have in the control of turning could be analysed in more detail (Fig. 3.6(b)). Once dry, the substance covered the whole of the tarsi, making it impossible for the insect to hold the surface. Blocking the front tarsi only for the duration of the experiment meant that the insects would not have sufficient time to extensively adapt to their new condition. Although the front legs were impaired they could still influence the direction of the body and its rotation, but the contribution was substantially reduced. This is because without being able to grip the substrate, the effectiveness depends only on frictional forces, and these in turn depend on normal forces with the surface. As the stick insect's centre of mass is positioned behind the metathorax, load on the front legs is small, hence so are these forces. The paint was removed by applying sufficient water to dissolve the paint again without mechanical intervention.

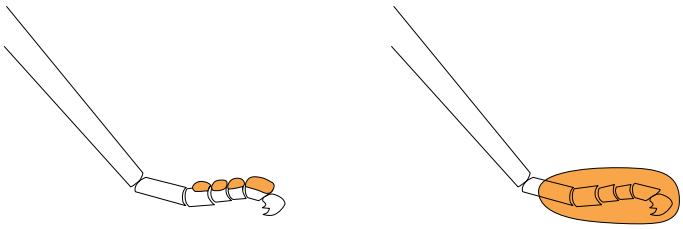
3.4.1 Methodology

Trajectories were recorded with a moving handheld video camera at a height of about 30 cm; the resolution of the camera (DCR-TRVHE) is 720x576 pixels at 25fps. Sequences were analysed with the visual tracking software detailed in section 3.3. A typical trajectory after processing a video sequence is shown in Figure 3.8(a). Three

[2]Bristol Fluorescent Paint, red UV Reactive water based 225gms(200ml)



(a) Photograph



(b) Schematic

Figure 3.6: Tarsus marks and schematic of tarsus blocked

(a) painting the tarsus without obstructing its functionality requires that the insect is moving until the paint dries. No paint should stay on the ventral side of the feet or on the claw. Note how the tarsus bends in both directions; on the left, the insect was climbing. (b) left: For marks used for tracking the paint was applied in less quantities and when the insect was in motion. By the time the paint dried, spots did not obstruct normal tarsus operation. (b) right: When blocking the front tarsus, the amount of paint covered the whole tarsus and the insect was not allowed to walk immediately afterwards.

marks on the body and one on each tarsus were followed (Figure 3.6(b)).

It was previously suggested that body extremity trajectories as well as the instant axis of rotation present a more comprehensive set of variables for study than the COM. Thus, points of interest along the body are the prothorax and the metathorax, whose velocities are $\vec{v}_P$ and $\vec{v}_M$ respectively. These velocities are relative to their own thoracic segment and to the body longitudinal direction. Velocities are represented in polar coordinates, i.e., $\theta_{\vec{v}_P}$ and $\theta_{\vec{v}_M}$ for the angles and $r_{\vec{v}_P}$ and $r_{\vec{v}_M}$ for the speeds.

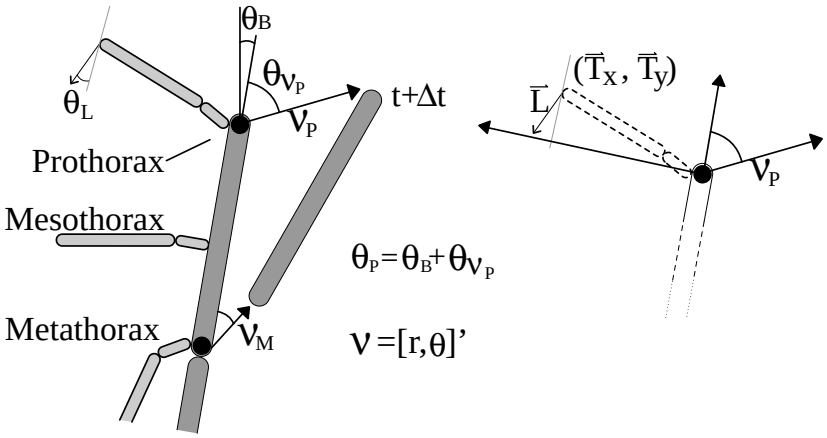


Figure 3.7: Notation of vectors tracking pro- and meta- thorax

Left: At least two different velocity vectors (speed and direction) are needed to describe body motion, these are relative to their own thoracic segment and body longitudinal axis. The body angle θ_B is in global coordinates and is zero when the turn initiates. Right: Tarsus position $\vec{T}$ and leg velocity $\vec{L}$ are also described relative to its own thoracic segment.

The angle to the target θ_T is in global coordinates and is constant for the whole of the turn. At the time the visual target is introduced, all angles are considered to be zero and augmenting thereafter as the body angle θ_B turns towards θ_T. However, the insect most probably calculates the instantaneous angle to the target relative to its own motion $\phi = \theta_T - \theta_B$. The angles $\theta_P = \theta_{\vec{v}_P} + \theta_B$ and $\theta_M = \theta_{\vec{v}_M} + \theta_B$ were introduced. The variables studied are shown in Figure 3.7. The target angle for all trials was normalised to one, the rest of the variables were adjusted accordingly. Additionally, turns were normalised to the same average time; this was 60 frames (2.4 seconds).

Section 3.3 explains the smoothing procedure that had to be carried out for studying changes in direction because of the limitations on camera movement compensation

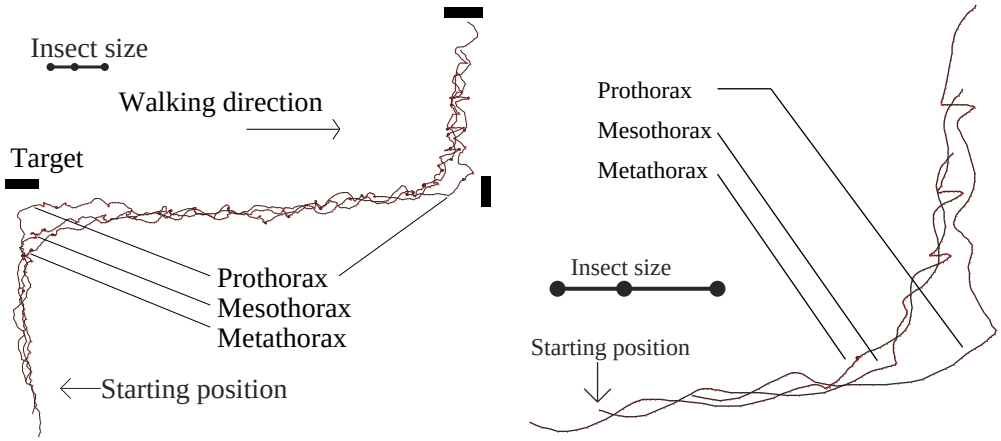
(section 3.3.1), variability in mark position (section 3.3), image resolution, and the oscillation of the insect's body. Direction of movement was calculated with respect to a point in time *n* frames ahead of the current position, typically 10 frames ahead. This lag removed noise and intrinsic oscillations of the stick insect, but tended to smooth fast changes in angle, particularly affecting those changing more abruptly, such as $\theta_{\vec{v}_P}$.

3.5 Results for Intact Insect

Figure 3.8(b) shows a typical left turn followed by one of the insects. Qualitatively it can be seen that the rear segments trace out a more curved path than the prothorax. Figure 3.9 shows, in the upper three plots, the angles followed by the body θ_B, the prothorax θ_P and the metathorax θ_M during a turn. For comparison, the three means are superimposed on one graph on the lower left. Despite the smoothing, it can still be clearly seen that at the beginning of the turn ($\theta_B \approx 0$), the prothorax direction θ_P changes within just a few time steps, and very early during the turn is aimed towards the target ($\theta_P = 1$). Remember that the target angle was normalized to one. It tends to overshoot the target during the second phase of the turn, particularly in turns larger than 70 degrees, in which the back of the body tends to rotate on the spot. It was also noticed that on certain occasions both front legs were lifted off the ground for a small period of time. During this time, the body's forward speed was almost zero, but rotation continued, resulting in a sharply curved trajectory at the beginning of the turn. Prothorax leg direction $\theta_{\vec{L}} + \theta_B$, relative to the target, during stance is also plotted in Figure 3.9. It can be seen that the front inner leg in particular moves towards the target and most probably pulls on that direction.

The metathorax, on the other hand, follows a smoother transition, similar to that of the whole body, i.e. θ_B. The speed $\vec{v}_M$ for this part of the body is low compared to the prothorax $\vec{v}_P$. In some cases, the prothorax moved at twice the speed of the metathorax, as shown in the lower right plot in Figure 3.9. During the initial transition of the turn, the hind legs are perhaps decelerated by the change in front leg direction; after one third of the turn they start accelerating again. In Figure 3.9 it is shown that on average the speed of the metathorax $r_{\vec{v}_M}$ is reduced by 60% at the slowest point, whereas the prothorax $r_{\vec{v}_P}$ is only affected by a speed reduction of 15%.

The IARs for the body were calculated at each moment during the sequence, for rotations larger than 1 degree. Figure 3.10 shows a normalized graph of accumulated IAR. Most IAR accumulated between the mesothorax and the metathorax, laterally



(a) Two turns, right and then left (b) Zoom of the second turn (Left)

Figure 3.8: Representative path: Intact stick insect

Example of the paths stick insects follow when attracted to a black vertical bar. A smoothed zoomed section of the last turn made is shown on the right, this represents a typical turn as analysed in section 3.5. Note how prothorax direction changes abruptly compared to other thoracic segments.

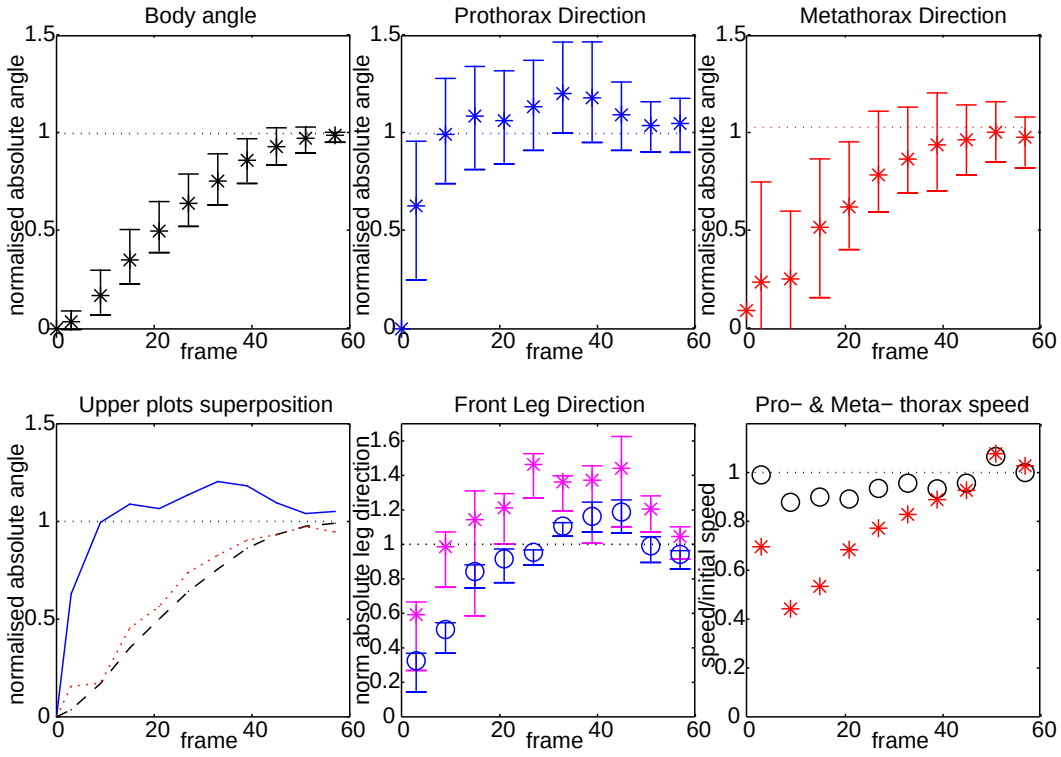


Figure 3.9: Body kinematics: Intact stick insect

Absolute angles normalized to the target angle $\theta_T = 1$ (unitless) and the bars indicate standard deviation. Top left shows progress of the body angle θ_B, top middle is that of the prothorax θ_P and the top right is the metathorax direction θ_M. Superposition of top plots is shown in the bottom left; metathorax θ_M with a dotted line; and the body θ_B with a dashed line. The bottom right shows changes in speed for the prothorax ($\circ$) and metathorax ($*$) relative to their velocity before the turn.The bottom middle shows the direction front legs follow relative to the initial heading, i.e., $\theta_B + \theta_L$; both legs are shown, inner front leg ($\circ$) and outer ($*$).

displaced from the body towards the position where the legs contact the ground. Having an average axis of rotation (AAR) close to the metathorax is in agreement with results shown for the speed of this segment, as the tangential velocity is proportional to the distance of the axis of rotation.

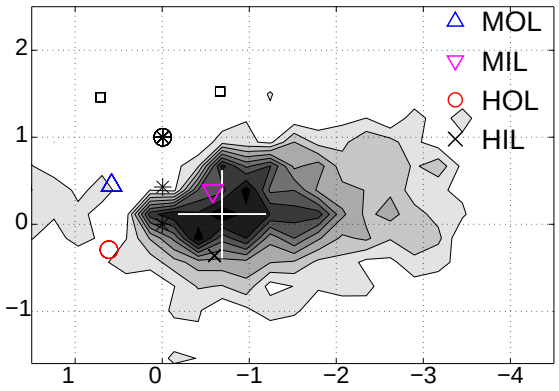


Figure 3.10: IAR: Intact stick insect

Cumulative plot of the instant axes of rotation (IAR). The contours represent the distribution of the IARs, the axes is the distance normalised to the distance from prothorax to metathorax. The white cross shows the average axis of rotation (AAR). The three black asterisks ($*$) represent the three thoracic segments and the prothorax (front) is encircled. All turns were properly transformed to the right before statistics were calculated. Middle 'M', hind 'H', inner 'I' and outer 'O' legs 'L'.

These data suggest that the specific movements of the stick insect's legs during turns results in the prothorax segment following mostly straight lines, pointing most of the time towards the target θ_P =1, whereas the mesothorax and metathorax tend to follow curves, with an AAR between the inner legs of the metathorax and mesothorax.

3.6 Experiments With Front Tarsi Blocked

Figure 3.11 shows two typical turns done by the stick insect when the front tarsi slip. The insect in this particular sequence is moving as shown by the arrow. Turns can be smooth like the first left turn (the prothorax following a curved path) or sharp like the second (the prothorax direction changing abruptly and early in the turn). Smooth turns take longer and require more space to get to the target. For the sequences analysed,

80% of turns were sharp instead of smooth, perhaps because the target was not far from the insect and thus an immediate adjustment in direction was required.

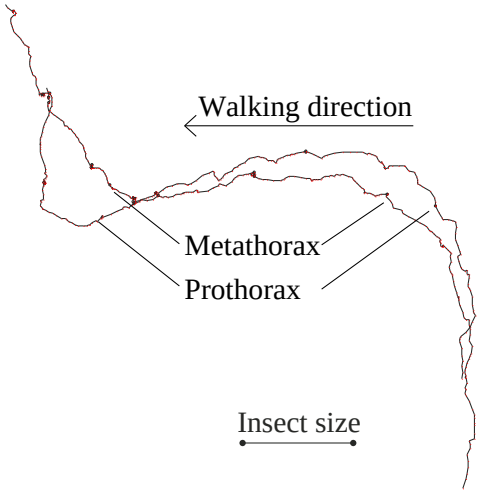


Figure 3.11: Representative path: Stick insect with front tarsi blocked

Representative example when attracted to a black vertical bar when front leg tarsi are blocked. Trajectories were either smooth (20%) or sharp (80%). No direct relation of turn type with the angle to the target was found.

In general, insects behaved more erratically for these experiments; they were more likely to stop in the middle of the turn and their speed was much slower. Insects tended to hesitate more between steps, particularly because the front legs did a series of short steps when they could not grasp the ground. Especially for the first presentation of the target, insects tended to create smooth curves, however, in contrast with the previous experiment (tarsi unblocked) this sometimes happened with subsequent targets as well.

The behaviour was analysed only for turns larger than 45 degrees because for smaller angles the front legs do not slip consistently and their contribution cannot be neglected. Figure 3.12 shows analogous results to those presented in section 3.5, except for the middle bottom figure which, instead of showing front leg trajectories shows body directions for smooth turns only. For most sharp turns (bottom left) the body pointed to the target when the prothorax was still moving laterally; therefore the prothorax direction ends at a slightly larger angle compared to that of the body. For smooth turns the prothorax direction only slightly deviated from the body and the metathorax direction, as shown in the middle bottom graph. The prothorax movement does not point consistently towards the target, in contrast with results shown in section

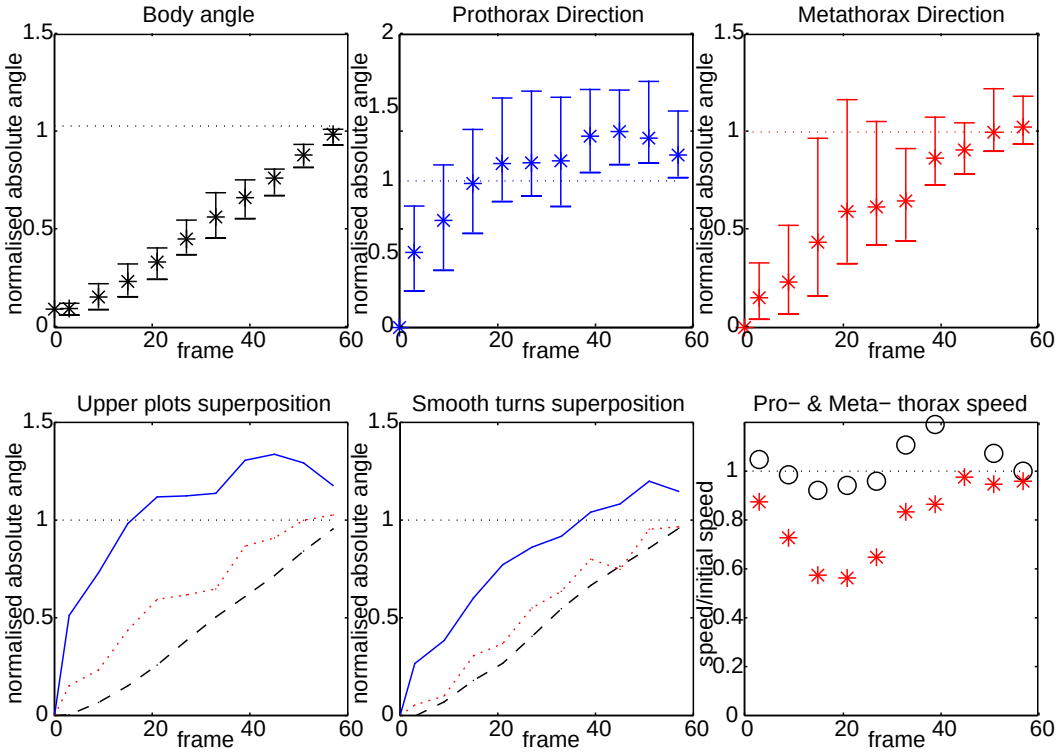


Figure 3.12: Body kinematics: Stick insect with front tarsi blocked

Statistics are from sharp turns, except for the middle bottom figure, which shows results from smooth turns. Absolute angles normalized to the target angle $\theta_T = 1$ (unitless) and the bars indicate standard deviation. Top left shows progress of the body angle θ_B, top middle is that of the prothorax θ_P and the top right is the metathorax direction θ_M. Superposition of top plots is shown in the bottom left; metathorax θ_M with a dotted line; and the body θ_B with a dashed line. The bottom right shows changes in speed for the prothorax ($\circ$) and metathorax ($*$) relative to their velocity before the turn.

3.5; this can also be seen in Figure 3.11. This difference supports the conclusion that normally the front legs pull the prothorax towards the target, but they cannot do so when the tarsi are blocked.

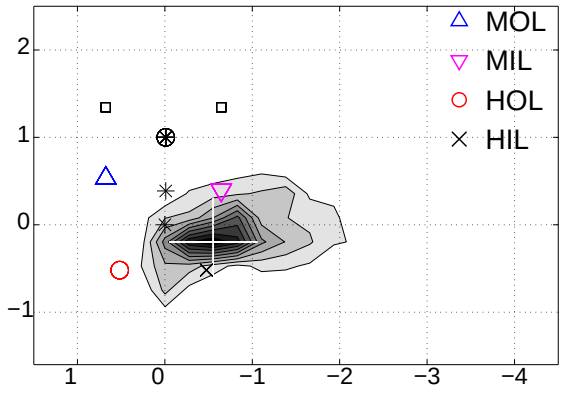


Figure 3.13: IAR: Stick insect with front tarsi blocked

Cumulative plot of the instant axes of rotation (IAR). The contours represent the distribution of the IARs, the axes is the distance normalised to the distance from prothorax to metathorax. The white cross shows the average axis of rotation (AAR). The three black asterisks (∗) represent the three thoracic segments and the prothorax (front) is encircled. All turns were properly transformed to the right before statistics were calculated. Middle 'M', hind 'H', inner 'I' and outer 'O' legs 'L'.

The AAR of all turns is shown Figure 3.13. This axis depends on the sharpness of the turn. In sharp turns the hind inner leg is often arrested in position, or even moved in reverse. The IAR in these situations is moved almost on top of the insect body.

Experiments on crayfish described in (Domenici et al., 1998) led these authors to propose that the correlation between inward angular acceleration and power stroke of legs could indicate that legs on stance at that time are contributing to turning. However, if the legs are in phase, their individual contribution to turning cannot be determined. Here a similar approach is followed to analyse the contributions of the middle and hind legs to turning. The technique focuses on changes in angular acceleration and speed of the body as each leg is lifted (changing from stance to swing at the posterior extreme position (PEP)) or placed down (changing from swing to stance at the anterior extreme position (AEP)). For instance, if one leg is strongly contributing to turning and it is suddenly lifted, then the body will decelerate during that transition. If all the legs followed a precise trajectory towards the target, i.e., contributing identically

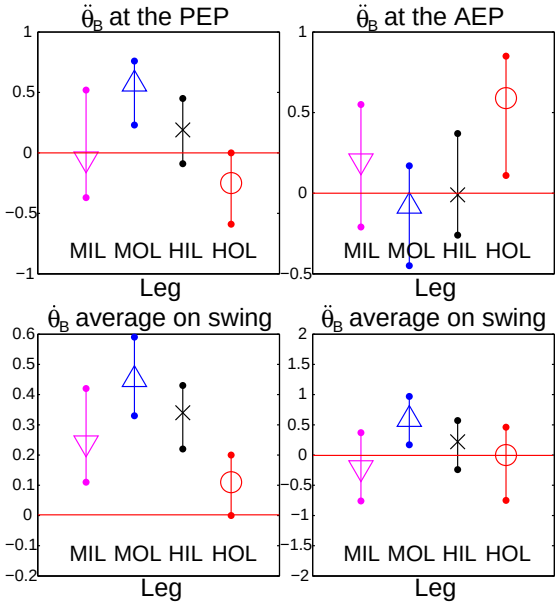


Figure 3.14: Swing statistics: Stick insect with front tarsi blocked

Accelerations and speeds of the body during turning corresponding to lifting or placement for each leg is shown; standard deviation is shown with lines terminated with dots. The two top graphs show the acceleration for each leg during transitions from stance to swing (PEP) and from swing to stance (AEP). The two graphs on the bottom show the average angular speed and average angular acceleration during turning while each leg is swinging (i.e. not contributing to the turn).

to the body trajectory, then it would not matter which leg was lifted, there would be only decelerations when lifted and accelerations when placed down, all with similar magnitudes.

It is assumed that two legs do not swing or hit the ground at exactly the same time, which was true for all sequences when front tarsi were blocked. This increases correlation between angular acceleration and leg contribution for individual legs. Furthermore, the number of legs contributing to turning for this analysis (n=4) compared to experiments on crayfish (n=8) further reduces uncertainties. The results are shown in Figure 3.14.

An increase or decrease in acceleration during the transition of a leg from stance to swing (PEP) or vice-versa (AEP) indicates whether the leg was opposing or contributing to turning. For instance, when the middle inner leg (MIL) swings the rotation of the body decelerates; it accelerates again when this leg touches the ground. The same happens for the hind outer leg (HOL) with a stronger effect. Hence, these two legs contribute to turning. The two graphs at the bottom show the average angular acceleration $\ddot{\theta}_B$ and average angular speed $\dot{\theta}_B$ of the body when the different legs were in the swing phase. Clearly the MOL is slowing down the rotation of the body when on the ground, as the angular speed and acceleration is maximized when this leg is swinging. It is important to note that turn contributions at different step phases were consistent for individual legs throughout, e.g., the MOL does not contribute turning at any point. The hind inner leg (HIL) moves little for sharp turns functioning as a pivot for turning, however, it is slowing down body rotation just as the MOL.

3.7 Analysis and interpretation of Turning

The individual leg trajectories for achieving the body trajectories described in sections 3.5 and 3.6 vary considerably on different thoracic segments. Moreover, individual leg speeds on either side of the turn vary greatly. Behavioural experiments show two distinct features for turning: First, the prothorax directs ‘itself’ towards the target in a seemingly straight line (3.5); second, without the front leg contribution, the body mainly rotates (3.6). The problem addressed in the following sections is how this complex pattern of leg behaviour and inter-thoracic contributions might be achieved with a biologically based control model.

Traditionally, turning in insects has been described and controlled by changes in step frequency or stride length. However, here these are replaced by speed and di-

rection of stance for the following reasons. Step frequency $f_S = T_S^{-1}$, depends on the stance period and the swing period[3], i.e., $T_S = T_r + T_p$. According to Graham (1972), T_p does not depend on walking speed for the tetrapod gait and turning. Therefore, step period directly depends on stance period T_r. Stance speed rather than stance period should be used to study leg kinematics because the latter cannot provide instantaneous information. Furthermore, step frequency alone cannot control turning.

Suppose a two-legged body moves by setting velocity $v_L = 0.25$ on the left, $v_R = 0.5$ on the right and swing phase takes $T_p \ll 1$. From those velocities, it can be calculated that the AOR is to the right at a distance equal to the distance between the tarsi. However, there are various possible step frequencies for these legs moving at the velocities defined above, i.e., for the same trajectory. If the transition from stance to swing is timed and legs can retract indefinitely, then $f_{SL} = f_{SR} = T_r^{-1}$. If for instance, step frequency depends on mechanical constraints, i.e., if after some fixed retracted position legs are forced to swing, then $f_{SR} = 2T_r^{-1}$ and $f_{SL} = T_r^{-1}$. Therefore, step frequency is biased by leg coordination parameters, whereas leg speed is more reliable to describe body trajectory in general. Furthermore, for more than two legs it becomes more unpredictable to analyse turning based on step frequency.

Therefore, leg speed contains more information about turning than step frequency. However, changes in leg speed require changes in leg direction for systems with more than two legs. Therefore, direction of stance needs particular attention as well. Consider a body with 'M' legs, where $M = \{2,3,4,\ldots\}$ and their position along the body is arbitrary. If contralateral leg speeds are changed without changes in stance direction, i.e., all legs on one side going at the same speed but different to the other right side, the turn will depend on friction at the tarsi. However, trajectories would be stochastic because friction is a statistical parameter that depends on normal forces. Consequently, turning cannot be controlled or studied reliably. Nonetheless, if the body rotates with no slip at the tarsi, which more closely describes insect locomotion, it can be demonstrated that leg speed and direction always changes.

Figure 3.15 illustrate the case when $M = 6$; the IAR is shown with the 'plus' mark. If the body shown rotates around this point, tarsi trajectories follow trajectories along an imaginary circumference with centre at the IAR. The speed of the tarsi is proportional to the distance to the IAR and the direction does not depend on the body. Note that at least $M-2$ legs will move at different speeds, i.e., at most two legs can move with the same speed. Furthermore, at most two stance trajectories can move parallel to

[3]Notation is after (Graham, 1972), subindexes referring to retractor (stance) and protractor (swing).

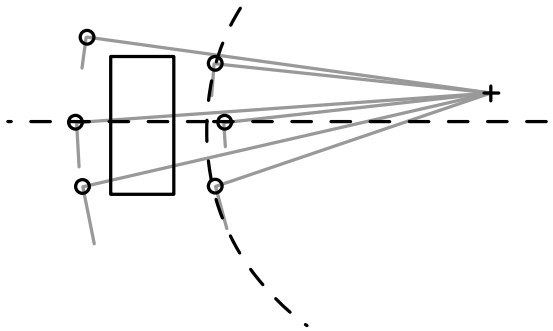


Figure 3.15: Leg direction and speed while turning

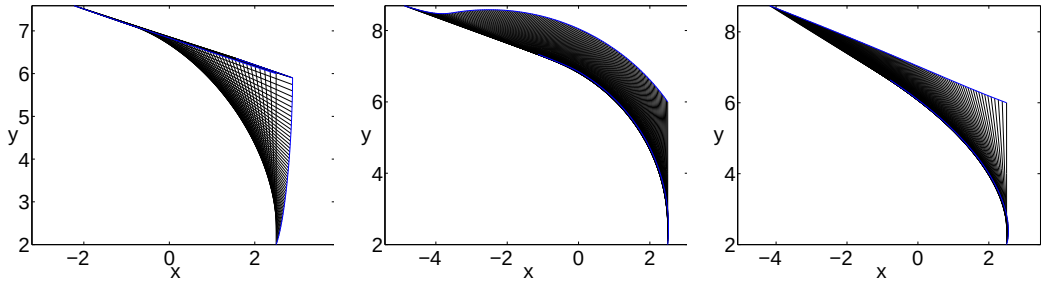
Leg speed and direction depends on the IAR, not on the body. Stance direction is perpendicular to the body only when the IAR is also perpendicular to the body and align with the tarsi. Two legs may have the same speed only when they are at the same distance to the IAR.

the body, only when the body, tarsi and IAR are aligned, i.e., at least $M-2$ legs will change direction. If two legs move at the same speed, they will not move parallel to the body. Note also that the IAR is variable and thus it is more likely for a given curved body trajectory to have all legs changing both speed and direction continuously. Therefore, these parameters, and not the step frequency and stride length, are more suited to study turning.

3.7.1 Body Trajectory Analysis

For every change in heading, it is always possible to find the IAR. Stance direction and speed depend on the IAR relative to the tarsus position. However, is the IAR a control variable or just a useful parameter to study? In other words, given the visual stimulus, does the insect's brain use a representation of the IAR to direct turning? This is unlikely, given the relationship between the prothorax direction and the target described in section 3.5, which is constant. Similarly, introducing the angle to the target ϕ as a variable to control directly based on the body bearing, $\theta_B = k\phi$, would produce curves in most cases. The gain of this equation, which would represent how closed the IAR is moved toward the body, does not affect the overall result. This is because the relationship between the angle to the target and the body angle is non-linear, as we shall demonstrate hereafter. An example for this strategy is shown in Figure 3.16(b). Therefore, the AOR for the stick insect is not a feasible variable to

control directly, rather it should emerge from other control mechanisms.



(a) global knowledge (b) angle control (c) equations in sec. 3.7.1

Figure 3.16: Problems with unbalanced opposing forces in middle and hind legs

Axes are x-y distance, i.e., the trajectory is seen from the top and the body is turning to the left. (a) Exact solution for thoracic straight-line following results in an unnatural turn and requires global information. (b) Controlling the angle to the target to zero, $\dot{\theta}_B = k\phi$. (c) kinematic model using model shown in section 3.7.1; prothorax, mesothorax and metathorax calculate translation and rotation.

The body motion, particularly that of the prothorax, has a strong relationship with the relative angle to the target. Therefore, the model presented here focuses on describing the motion of this thoracic segment. Assuming a constant body height, the insect body position can be specified by its projection onto the ground plane (2D). Then, leg kinematics to create such movement can be more easily proposed.

3.7.1.1 Prothorax

Since it was assumed the IAR is not a variable to control, it will not be introduced as a control variable. The metathorax speed $r_{\vec{v}_M}$ was not found to be zero and the IAR was positioned to one side of the body, hence, $\vec{v}_M \neq \vec{0}$. The prothorax velocity moves in the direction of the target, i.e., $\theta_{\vec{v}_P} = \phi$, but this is not sufficient to induce body rotation. Nonetheless, if it were to rotate by $\Delta\theta_B$, after some time $\Delta t \ll 1$, a line at an angle $\Delta\theta_B$ from the new prothorax position to an imaginary rotation point aligned with the body can be found. This line is illustrated in Figure 3.17 and velocity vectors along this line $\vec{B}_\eta$, are described by equation (3.7),

$$\theta_{\vec{B}_\eta} = \arctan\left(\eta \tan\left(\theta_{\vec{B}_{\eta=1}}\right)\right) \tag{3.7}$$

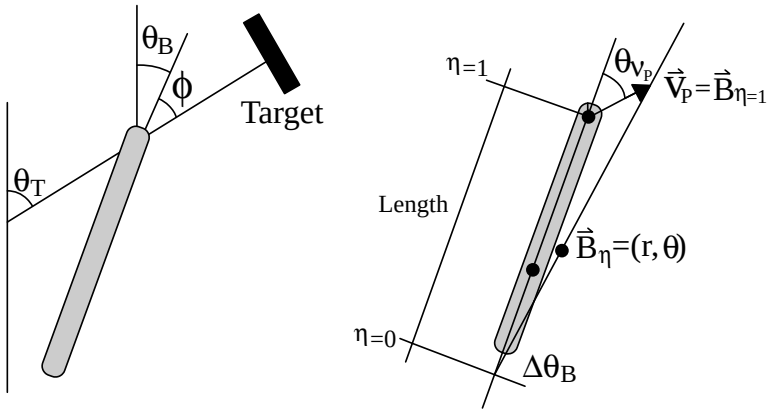


Figure 3.17: Variables related to the visual target

Left: θ_T is the angle to the target, θ_B is the body angle and ϕ is the relative angle to the target. Right: The prothorax is translated by $\vec{B}_{\eta=1}$ and rotated $\Delta\theta_B$ around $\eta = 0$. $\vec{B}_\eta$ is the translation followed by other points along the body.

where $\eta = \{0...1\}$ is the relative position between the intersection point and the prothorax. This solution is incremental, and requires $\Delta\theta_B \ll 1$. It is straightforward to verify that for $\eta = 1$ the direction of the prothorax is that of the relative direction to the target $\theta_{\vec{v}_P} = \phi$. This is the condition that corresponds to experimental results shown in section 3.5.

There are only two points whose solution is linear: $\eta = 1$ (prothorax) and $\eta = 0$. The latter however is variable and not related to any thoracic segment. Furthermore, the body angle, given by equation 3.8, appears to be equally unlikely to be a control variable.

$$\Delta\theta_B = r_{\vec{v}_P} \sin(\phi)/Length \tag{3.8}$$

If these variables were used, transforming this model of body motion into leg trajectories $\theta_{\vec{L}}$, is fairly straightforward: if $\vec{T}$ is the position of the tarsus relative to the coxa, and that thoracic segment is to move by $\vec{B}_\eta$ with the body rotation $\Delta\theta_B$, then it follows that $\vec{L} = [-\vec{T}_y, \vec{T}_x]\ \Delta\theta_B\ -\vec{B}_\eta$. However, this equation for $\vec{L}$ implies that all legs in a kinematic model need to calculate at each point how much the body needs to rotate, and the middle and hind legs need to know η. Furthermore, the hind legs direction will depend not only on ϕ, but on $\arctan(\eta \tan(\phi))$. A kinematic example using this approach is shown in Figure 3.16(c); all thoracic segments calculate translation and

rotation. The trajectory is as expected.

However, the procedure can be simplified so that it is not necessary to explicitly determine $\Delta\theta_B$ or η. Implemented in a dynamic model, equation (3.7) could be executed just for the prothorax, i.e, $\theta_{\vec{v}_P} = \phi$. Additionally, the back of the body could account for the rotation $\Delta\theta_B$ without explicitly following equation 3.8. This further simplifies calculations for leg trajectories in the prothorax, because if they no longer need to compute $\Delta\theta_B$, the equation for front leg trajectories becomes simply $\vec{L} = -\vec{B}_\eta$, i.e. front legs need only to move in direction ϕ with respect to the body, $\theta_{\vec{L}_{Pro}} = \theta_{\vec{B}_{\eta=1}} = \phi$.

Note that there are an infinite number of solutions for the condition $\theta_{\vec{v}_P} = \phi$. However, of various kinematical models tested that described straight lines for the prothorax, the best one for simulating the body trajectories of section 3.5 is described by equation 3.7.

3.7.2 Body Rotation Analysis

Normally, when the body rotation is sharp, the hind inner leg is almost arrested near the AEP, and the body rotates around this axis. Based on Figure 3.15, Figure 3.18 on the left shows how each middle and hind leg should move to contribute to this specific rotation: specifically, the MIL should move mostly sideways, and the HOL should move front to back. The relationship between the angle to target and the mesothorax or metathorax motion is non-linear. Therefore, there is no evidence that middle and hind legs direct their movement in an obvious manner towards the target.

3.7.2.1 Metathorax

If no active lateral force is needed for the metathorax behaviour, then the hind legs need only to contribute based on a contralateral difference in speed. Control could consist of altering the average speed $r_{\vec{L}_{meta}}$ of both hind legs by the same, but opposite, value Δr_H, Figure 3.18 (right). If no other legs were involved, the distance to the IAR from the hind inner leg would be $IAR_{meta} = l_{meta}(r_{\vec{L}_{meta}}/\Delta r_H - 1)/2$; where l_{meta} is the distance between the hind legs. Note that this simple control for hind legs could result in the hind inner leg being arrested ($\Delta r_H = r_{\vec{L}_{meta}}$), or going in reverse ($\Delta r_H > r_{\vec{L}_{meta}}$). This control scheme is consistent with the body acceleration effects of lifting and placing the HOL as shown in Figure 3.14, although it is not clear why when the HIL is swinging, there is a (small) angular acceleration of the body (i.e. why this leg

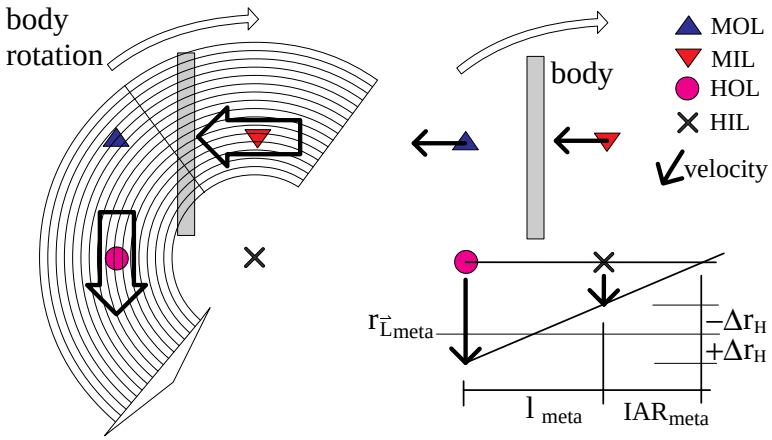


Figure 3.18: Ideal leg directions if IAR is close to the HIL

Left: Ideally, if the body were to rotate around the hind inner leg, the MIL and the HOL would have specific orthogonal trajectories to follow. Right: Graphical representation of leg's active role. The hind legs' average speed $r_{\vec{L}_{meta}}$ is changed by $\pm\Delta r_H$ resulting in an IAR_{meta} that depends on the distance between hind legs l_{meta} and the resulting difference in hind leg speed.

seems to slightly oppose the turn). However, Figure 3.18 is just an extreme state for illustration and in practice other factors will affect the exact nature of leg contributions.

3.7.2.2 Mesothorax

First of all the MIL contributes to turning whereas the MOL seems to oppose turning with a much stronger effect than that for the HIL (Figure 3.14). This could be explained if both middle legs produced a perpendicular force to the body (Figure 3.18). Under this condition only the MIL would contribute appropriately to the rotation, while the discrepancy between the trajectory of the outer leg and the ideal would oppose the direction of the MIL and the HOL. Active pulling of the MIL has been suggested for some turns on the cockroach at low speed (Mu and Ritzmann, 2005).

In summary, the body rotation can be achieved by having the metathorax rotating by means of difference in speed on either side and by having the mesothorax moving sideways. This is easily translated into joint control by the observation that the BC joint is particularly related to back and forward movement of the leg, and the FT joint is related to lateral movement of legs (see 2.2). By introducing differential activation of the BC joints in the hind legs, their relative speed can be easily controlled. Similarly, introducing a bias to the FT joint in the middle legs will produce the required lateral movements.

3.8 Discussion

In this chapter, it was shown that the stick insect is capable of directing itself towards a visual target. The highly correlated trajectory that the prothorax follows with respect to this target suggests two hypotheses. Firstly, that the insect can measure the relative angle to the target ϕ by visual processing. Secondly, that it can use that angle to direct the prothorax in a straight line towards the target. Additionally, the role of the front legs in the processing of the ϕ is of major significance. This was based on the different trajectories the prothorax follows depending on whether the front tarsi is blocked or not. The direction of the prothorax in Figure 3.12 does not have a strong correlation with the target as that shown in Figure 3.9.

Behavioural experiments with front tarsi blocked showed that individual middle and hind legs behave differently while turning. The middle inner leg and the hind outer leg induced body accelerations suggesting that these contribute to the turn, whereas the

middle outer leg and the hind inner leg oppose it. This opposing motion is not directed in an opposite direction to the turn, as with the cockroach while running (Jindrich and Full, 1999b). The angular body deceleration by the MOL and HIL indicate that their contribution to turning is not as efficient as that of the MIL and HOL. Results indicate that individual leg kinematics induce turning, but each in a different direction.

A contribution of this thesis mentioned in section 1.5 is an improvement in the methodology for analysing body and leg kinematics. In section 3.2 it was mentioned that usually variables related to the COM are reported. The COM in the stick insect is positioned close to the metathorax, however its trajectory is not correlated to the angle of the target as that of the prothorax is (Figure 3.9); nor is that of the mesothorax. Furthermore, the angle of the body (heading) and translation of points along the body are not correlated either. For instance, the prothorax sometimes moves perpendicular to the body axis (Figure 3.8(b)). Additionally, the position of the IAR relative to the body provides more information about body kinematics than the curvature, which is only the distance to the IAR from the body.

3.8.1 Single Leg Controller Implications

Behavioural experiments suggest the presence of a continuous negative feedback controller with a calculated angular velocity reference as mentioned in section 2.7, i.e., leg's internal intention is under negative feedback and not only active for small disturbances. The negative feedback is at the level of individual joints and tarsi trajectory because the former depends on the latter. Support for this hypothesis can be found in both experimental settings, with intact stick insects and with front tarsi blocked. In the first set of experiments, it was suggested that front legs play an active role for turning because without tarsi the prothorax direction no longer points towards the visual target. The role of influencing middle and hind legs would be very difficult, if not impossible, to provide by front legs if these do not oppose large forces continuously. Furthermore, in the second experiment, if the middle and hind legs were only positive feedback-based, after some time turning they would continue to support that motion. However, different legs appear continuously to influence others in the same manner for the whole turn, indicating a continuous negative feedback mechanism sustaining its own direction. Directions that each leg appears to be maintaining do not seem to be in agreement with their current motion, i.e., they do not appear to be under positive feedback. It is assumed that the negative feedback mechanism in joints is influenced

in these experiments by visual orientation, however, we are also assuming that the mechanisms that govern turning are similar that those controlling other behaviours.

Tarsi do not slip, thus all joints in the body move in synchrony when the body moves towards the body. The Analysis presented in section 3.7 shows that to reproduce the trajectories of the insect, legs would need to behave according to specific functions. However, particularly for middle and hind legs the relationship to the target is non-linear. It is known from neurophysiological data that legs cannot use specific states from other legs to move accordingly. The need of a positive feedback mechanism is inferred based on the lack of information shared between legs. It is hereafter assumed that a positive feedback and negative feedback mechanism should be simultaneously present in the single leg controller. The negative feedback is suggested because of the need to have continuous correction of joint velocity, and the positive feedback because shared information among legs is very limited.

The reader might wonder why the positive feedback in middle and hind legs does not suffice to explain the behaviour of the intact animal. In fact, conclusions presented in (Rosano and Webb, 2006) suggested that middle and hind legs did not require other mechanisms. Nonetheless, the contrary has been suggested in (Rosano and Webb, 2007) and it will be later shown in subsequent chapters. These hypotheses will be tested using various models against behavioural results found for the stick insect.

3.8.2 Thoracic Differentiation

Experimental results presented in this chapter clearly highlight the role of the prothorax in the control of turning. It was also demonstrated that the mesothorax and metathorax contribute to turning mainly by rotating the body. Furthermore, the analysis presented in section 3.7 indicates that each thoracic segment could contribute to turns in different ways, with the angle to the target influencing mainly the FT joint in the mesothorax and the BC joint in the metathorax. Therefore, middle and hind legs appear to use the angle to the target ϕ as a tendency and not explicitly for the calculation of stance. It is possible that the insect changes strategy when front legs influence for turning is not present. This possibility was reduced by having the front tarsi blocked only shortly before the experiment and removed soon after, thus reducing the likelihood of adaptation. Further experiments need to address such possibility.

This proposed thoracic differentiation for the intact insect depends on the properties of the single leg controller discussed earlier. If middle and hind legs did not have the

positive feedback mechanism, they would have to calculate trajectories that do not show a clear kinematic constraint as with those for the prothorax in experiments with intact stick insects. Additionally, without the negative feedback mechanism in front legs, it would be more complicated to replicated the correlation found between the prothorax and the target. This would be further supported in chapter 5.

3.8.3 Open questions

This section shows what the insect can do and the hypotheses about how it might do it. At this point it is useful to go back to questions posed in section 3.1.

What and how complex is the information sent from the brain? It should be remembered that the behavioural experiment first required inducing stick insects to first start walking before introducing the visual stimuli. Furthermore, the information sent seems to include the angle to the target ϕ and not just a simple tendency to turn. Therefore, the brain sends the intention to walk (speed), and the relative angle to the target.

How do legs process information from the brain? It is clear that front legs are capable of directing the prothorax towards the target, therefore, the direction of stance must be corrected accordingly. Similarly, mesothoracic and metathoracic legs seem to change the direction towards the target, although it is not clear if the angle to the target is used or a discrete signal such as ‘strong’ or ‘weak’ rotation. Nonetheless, it seems that all respond actively to a command from the brain that indicates a particular heading it wants to reach. Therefore, this could be the answer to question 3, i.e., all legs respond actively for the control of turn.

However, there are still unanswered questions, e.g., whether leg coordination controls turning or emerges as a secondary effect. The question regarding what commands control leg direction is still speculative. Herein it is suggested that legs actively change direction of stance and simultaneously comply with other legs, but this experiment does not highlight details of the underlying mechanism.

3.8.4 Conclusions

Legs are bound to change direction and speed during stance while turning. Furthermore, it is important to note that this condition is not restricted to the stick insect; it is an intrinsic property of legged systems. In addition, the technique used mainly for studying biological joints proved to be insightful for legged body locomotion.

Behavioural experiments indicate that the stick insect measures the angle to the target and passes this information to the thoracic ganglia. These in turn control individual legs to change their direction of stance to direct the body towards the target. Front legs influence most of the turn and as a result the prothorax moves in a straight line towards the visual objective. The hypothesis drawn is that middle and hind legs seem to play a more passive role, nonetheless, they could too actively contribute, mainly by rotating the body.

Additionally, we suggest that single legs continuously oppose external and internal forces in order to sustain their own motion. Nonetheless, no slippage occurs and thus all joints are still mechanically coupled. Even if both negative and positive control mechanisms were present, it is not clear how these should be combined. This however, should be similar if the target is in front and the resulting motion is straight walking.

It is clear that the results of the rest of the thoracic roles for turning would not be possible without single leg controllers. However, note that the model for the single leg that solves forward walking should contain the necessary elements to direct stance and to be influenced in particular joints. Three important hypotheses from this chapter should be included when solving for straight walking as well:

- The angle to the target can be processed visually and passed to the thoracic ganglia.
- Legs are always controlled with negative feedback because they constantly oppose the movement of others.
- Thorax segment contributes with specific roles for turning.

Chapter 4

Subordinated Single Leg Controller for Walking

The last chapter (chapter 3) finished with some unresolved questions. Nonetheless, previous chapters offered a more comprehensive understanding of the necessary elements the single leg controller needs to incorporate for the control of walking. Before describing the proposed simulation model, at this point it is necessary to recapitulate on the elements which are needed to explain the walking behaviour, and which will be incorporated into the model.

The control of walking assumes that legs are not centrally controlled for gait coordination or stance trajectory orchestration (chapter 2). The architecture is decentralised in six interconnected legs whose shared information is limited. However, there are signals send to all pair of legs, for instance, walking activation or visually processed information (chapter 3). Furthermore, there has to be the possibility of commands being processed differently in each thoracic segment.

The turning behaviour cannot be explained by means of positive feedback alone, furthermore, the role of negative feedback is significant in all legs. The presence of a continuous negative feedback controller is supported by previous work, e.g., (Bartling and Schmitz, 2000) (section 2.6) and by the results presented in chapter 3. In addition, the active role of legs is further supported by concurrent research by Dürr and Ebeling (2005).

The leg controller is divided into two parts, the swing control and the stance control, their implementation is described in section 4.1 and section 4.2 respectively. The swing controller poses a less challenging control problem than the stance controller because the latter is mechanically coupled with other legs through the ground. The leg during

swing does not directly influence the direction of the body, only the touchdown position affects the duration of stance. Therefore, from the stance-swing transition (PEP) to the swing-stance transition (AEP) tarsus trajectories have no effect on body kinematics. In section 4.1 the swing controller implemented in the robot simulation is described.

Much attention was dedicated to the control of stance because it was the step phase more relevant to the hypotheses presented herein. A variable that allows legs to comply with external forces (positive feedback) is introduced but, most importantly, it is combined with the ability for each leg to sustain its internal intention (negative feedback). The controller behaviour is difficult to analyse analytically once implemented in a hexapod. Therefore, a simpler system is introduced in section 4.2.2 to show how the variables affect motion.

The transition between step phases is based on basic local rules proposed by Cruse (1990), resulting in stable leg configurations without having a central controller. This approach has been implemented successfully in countless experiments, nonetheless, the implementation followed for this model relates more to the control of turning; this is explained in chapter 5.

To summarise: in this chapter, a single leg controller based on key elements known to be present in the stick insect is proposed. These elements are based on information presented in chapters 2 and 3. To test these and future hypotheses a dynamic robot simulation was programmed; this is briefly described in section 4.3. Results are presented in section 4.4 and their performance and implications are commented in section 4.5.

4.1 Swing Controller

The directions of tarsi need to change when the leg is on stance in order to direct the body. It was proposed that the angle to the target ϕ is calculated in the brain and then passed to the thoracic ganglia. The trajectory followed by the prothorax indicates that the angle ϕ is constantly used to direct the body. This direction needs to be maintained based on current body and joint positions.

An alternative solution to control legs on stance is by specifying the AEP and the PEP (Cruse and Saavedra, 1996; Kindermann, 2001). However, this is not feasible, given the strong correlation between the front legs and the visual target angle. The direction to the PEP will no longer be parallel to the body and it would require updating the PEP according to the target and the previous AEP, i.e., PEP(ϕ, AEP). In other

words, interneurons at the ganglia would need to process ϕ in such a way as to produce an imaginary PEP that continuously attracts the tarsi. Dürr and Ebeling (2005) report changes in direction during an ongoing stance, therefore, to control stance direction with the PEP, it would be necessary to update the PEP based on the current tarsus position as well. This requires forward kinematic computation for the tarsus, to create the target PEP. However, the PEP would have to be represented in Cartesian coordinates as well, because while turning the BC, or any other joint, does not relate to the PEP as in straight walking. Alternatively, a PEP could be estimated based on joint angles, however, the relationship between ϕ and leg joints would be non-linear and more difficult to adjust continuously. Additionally, the PEP strongly depends on coordination influences and therefore its final position is uncertain. The PEP by definition is the extreme position at the end of the stance, however, if this point is uncertain, it becomes impractical to use this variable for reference or control purposes.

Similarly, the AEP is not explicitly calculated but results from the intersection of the joint angles and the ground surface. As previously mentioned, the ground is not expected to be found at any position (Dürr, 2001). Instead, at each protraction the leg stops only when the right signals indicate it can support the body. Therefore, the AEP cannot be accurately estimated beforehand for irregular terrain. Furthermore, without an end-reference (PEP), the AEP bears no meaning for stance direction. As previously mentioned, stance direction can change after the AEP, indicating that this position is not necessary for directing stance.

It has been noted that some legs could actively influence turning and others just follow passively (Domenici et al., 1998); consequently, changes in PEP and AEP could be a secondary effect[1]. Experimental results presented in chapter 3 support this idea, in which the leg controller estimates stance direction based on current joint positions. However, while shifts of the PEP might be a entirely secondary effect, shifts on the AEP are less likely to be a secondary effect because the AEP is not influenced by other legs as the PEP is.

Consequently, the AEP and PEP are useful stepping feature parameters that nicely describe overall single step trajectories. However, these positions are unlikely to control stance directly.

[1] These experiments were on crayfish locomotion.

4.1.1 Swing direction (AEP)

The AEP is not a feasible control variable for stance, however, it cannot be as easily explained as a secondary effect as the PEP. The AEP does not necessarily affect stance trajectory because we are assuming there is no end-reference as a control variable and the leg can control its direction based on current joint position. However, it does affect the time legs can be on stance. For instance, if the tarsus is to move towards the body it is better if it initially lands (AEP) away from the body, because otherwise leg motion would terminate prematurely. Therefore, a good positioning according to the next stance direction improves coordination.

It was decided not to implement the swing controller based on known biological data and models because herein it is assumed that the only influence swing has for walking is by defining the AEP. The swing controller is activated by a step selector (see section 5.1) with a BC_{AEP} and FT_{AEP} joint targets. Setting these two joints indirectly sets an AEP, however, the latter further depends on when the CT joint drives touchdown. As described below, this trajectory for getting to the AEP was not entirely neglected; however, it was empirically programmed. A more biological solution could be sought in future work, but it is beyond the scope of this thesis.

4.1.2 Heuristic Swing Controller

The swing phase was initiated by increasing the CT joint, which moves the leg up. Initially the CT changes its velocity, however the tarsus remains touching the ground for some time. During this offloading transition, the FT joint was moved opposite to the CT joint, and the BC joint passively followed external forces. These conditions were introduced to avoid perpendicular forces on the ground. As soon as the tarsus was off the ground, the final positions for the BC and FT joints were calculated according to the next leg direction; this corresponds to a possible AEP.

The FT joint was set to two possible values depending on the leg position relative to the turn. The outer leg, i.e., the leg about to push away from the body during stance, was given the value FT_{AEP} = -100. If the leg was in the inside of the turn, i.e., its future stance direction was towards the body, this value varied according to the following linear equation: $FT_{AEP} = -100 + 1.3\phi$. The BC joint was calculated using the following equation: $BC_{AEP} = 0.5\phi - 60$. The estimated final position produced by these equations represents a lateral shifting of the AEP, particularly inner legs, to the turn. This was sufficient to improve inter-leg coordination. These values were given to

all legs that received a stance direction command other than zero. The performance of leg coordination was not sensitive to variations on these values.

As soon the tarsus was off the ground and the estimated final position for the BC and FT joint was calculated, the leg was moved in two phases, an upward movement and a downward/searching movement. The transition from the former to the latter was triggered by the CT joint reaching a threshold angle of 80 degrees. During the whole of the swing the BC joint was proportionally controlled to get to the BC_{AEP}. However, the FT joint during the upward swing movement was proportionally controlled to get to -60 and during downward swing to FT_{AEP}.

Swing was terminated when the leg was going down and the load on the leg was large.

4.2 Stance Controller

One of the most challenging tasks is to orchestrate leg joints during stance phase. When many legs are attached to the ground, they are mechanically coupled through the substrate. The dynamics of such a system is highly complex and difficult to control. Many studies have been carried out in insects to find out what control strategies insects use, however, results suggest that rather than using a single control strategy, insect strategies are context dependent. Not only do joints respond differently to stimuli when insects walk or stand, changes are also found when walking on different surfaces.

The actual control of legs during stance phase is not fully understood. Mechanical interaction between legs and differences in behaviour in response to various experiments make it difficult to isolate the core of the control mechanism. Alternatives include PD position control, velocity control and hybrid position/force controllers. Nonetheless, most solutions agree that legs are not centrally controlled and therefore it is necessary to have cooperative mechanisms. It has been suggested that insects solve the problem of compliant motion by using positive velocity feedback control. However, as mentioned in section 2.6, a negative feedback mechanism is always present. Furthermore, behaviours where leg muscles need to oppose external forces are numerous, including protractor muscles for climbing, depressor muscles for stepping up or flexor muscles while turning. The latter is analysed in chapter 3 and it was suggested that direction of all legs are under negative feedback. Consequently, the hypothesis is that legs require the following characteristics:

- Respond to external disturbances according to a velocity negative feedback
- Move following a specific direction given by the brain
- Respond with a positive feedback to coordinate with the rest of the legs

Therefore, legs should respond positively to inter-leg forces to comply with other legs and simultaneously should respond negatively to external forces and to maintain its own motion. The distinction however is not trivial, as there is no central controller, and both forces look alike to individual legs. Bartling and Schmitz (2000) proposed a positive velocity feedback switching to a negative controller in response to high accelerations. The role of the negative feedback is therefore relegated to unexpected disturbances and joints are assumed mainly to follow passively. Alternatively, in (Schneider et al., 2005c, 2006) the controller's positive feedback was switched off when joints were not contributing to the mechanical power of the leg, i.e. joints were compliant when the joint power was negative. Nonetheless, in section 3 it is shown that each leg tries to follow its own direction, i.e., they seem to be under constant negative feedback. Therefore, this suggests that individual legs cannot really distinguish between forces caused by other legs and external forces.

4.2.1 Single Leg Controller

Using positive feedback as the central controller implies that the total combination of external forces would eventually result in the desired movement with little intervention. Effectively, positive feedback tries to minimise changes in the energy of the system by multiplying external forces instead of opposing them. This situation is particularly well suited for repetitive movements or for trajectories that do not tend to change voluntarily, i.e. energy conservation situations. However, mechanical systems also require to output energy to do some external work, e.g. to change inertia or to move external objects. In these situations, positive feedback alone is not sufficient to complete the task (Schneider et al., 2005c).

Neither leg trajectories, nor the body, follow a consistent trajectory for the control of turning. Consequently, the system requires a large energy input to start walking and to correct continuously direction. The behaviour to be simulated requires cooperation between legs as well as constant correction for trajectories. Additionally, it is not optimal to correct angular speed continuously based on a positive feedback controller.

Instead, the hypothesis for the proposed model is to have the intended leg direction biased by external forces and the negative feedback controller as the central controller.

One important requirement for positive feedback mechanisms is to detect the direction of the external force. This requires either a force sensor or a non-stiff joint. In section 2.2 it was mentioned that stick insect can measure stress on the exoskeleton, however, joints in the stick insect are also non-stiff. However, one possible solution is to have a spring system between the motor and the segment simulating the non-stiff situation. A larger deviation can be measured with a position sensor instead of using a force or strain sensor. This is shown in Figure 4.1. The motor moves the spring case, which if free of external forces, maintains the final actuator positioned at the centre. This configuration has been implemented in recent simulated versions of the Tarry robot and future studies on positive feedback control architectures in the real robot.

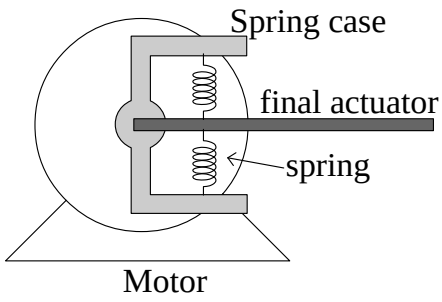


Figure 4.1: Spring system representation implemented in joint motors

The motor has direct control only on the spring case, the position of the final actuator depends also on external forces.

4.2.1.1 Internal Intention: Negative Feedback Mechanism

Controlling directionality of the tarsus in a 3D space requires joints in each leg to have information about other joints within the leg. This local assumption has been implemented on the CT joint controlling height, e.g. (Kindermann, 2001; Schmitz et al., 2001; Schneider et al., 2006). Therefore, it is equally plausible that the BC and the FT joint have access to similar joint information within the same leg. Furthermore, inter-joint coordination has been demonstrated in (Bucher et al., 2003; Cruse and Bartling, 1995; Brunn, 1998).

The first step is to have a function that calculates the direction of the tarsus move-

ment based on the current position. We propose to calculate joint velocities with the following equation,

$$\dot{\vec{W}}_{sp} = \begin{vmatrix} \dot{\alpha}_{sp} \\ \dot{\beta}_{sp} \\ \dot{\gamma}_{sp} \end{vmatrix} = [J(\vec{T}(\alpha,\beta,\gamma))]^{-1}\vec{L}(r_{\vec{L}}, \theta_{\vec{L}}, dz_{\vec{L}}) \tag{4.1}$$

where $J(\vec{T})$ is the Jacobian of the tarsus position, $r_{\vec{L}}$ is the desired leg speed, $\theta_{\vec{L}}$ is the direction and $dz_{\vec{L}}$ corrects for the local height $h_{\vec{L}}$. The local height was fixed and the equal for all legs. On the left, the desired increase in position (velocity) is represented by the subindex sp for set-point. The velocity the tarsus must follow is given by $\vec{L}$, which is the ganglion or brain command. The Jacobian processes the internal state of the leg and computes the next movement for the joints. The term $J(\vec{T})\vec{L}$ can be simplified with an artificial neural network, making it feasible to be calculated by a biological system. This is demonstrated in chapter 6.

The internal model uses the position of the leg, not the velocity, being based on that feedback signal, it partially resembles a position controller. The current angular positions and the inverse Jacobian are necessary to estimate the next movement. However, the input $\vec{L}$, and the output $\dot{\vec{W}}_{sp}$, are velocities, therefore, it is more precisely described as a velocity controller. The angular velocity is accumulated and the resulting position is controlled by the joint. Therefore, it is a nested hierarchical controller, with the position controller in the inner loop and the velocity controller in the outer loop. Note that the position controller is not represented in 4.1. Controlling velocity instead of position is prone to larger deviations in trajectories. Even if joints were stiff, the inner controller, using motors or muscles, would take time to get to the desired set-point. Furthermore, steady state error is rarely zero, particularly for higher order non-linear systems. Another complication is that even if there were a central controller, it would be unfeasible to orchestrate all legs by setting the only possible global variables, $r_{\vec{L}}$ and $\theta_{\vec{L}}$. For these reasons, equation 4.1 as it stands is likely to fail.

One solution to this problem could be to accumulate the error and compensate for that in the trajectory, i.e., introduce an integrator for the position based on velocity deviations. However, a simpler initial alternative solution could be to use the velocity controller in an open loop condition while the inner position controller remains active. The hierarchical nested controller becomes a cascade controller. Neither velocity nor current position is fed back into the Jacobian. This scheme ignores external forces and internal joint deviations, and tries to control the ideal direction given by the ganglion or

brain. However, the controller maintains the position-based negative feedback mechanism. The following equation represents the open loop velocity by using the set-point to compute the internal state (Jacobian) instead of the real position, which will in fact be deviated.

$$\dot{\vec{W}}_{sp} = [J(\vec{W}_{sp})]^{-1}\vec{L}(r_{\vec{L}}, \theta_{\vec{L}}, dz_{\vec{L}}) \tag{4.2}$$

The difference from equation 4.1 is that the joint set-point is used after the accumulator and not from the real position. To visualise how this equation is implemented and where the negative feedback is, Figure 4.2 shows the control block diagram. The open loop is on the left because it does not consider external influences. However, in the loop on the right, the servomotor (SM) or the motoneurons are to control the position using negative feedback. The world dynamics are represented by G. The role of the negative feedback is to sustain the internal intention that is specified by the Jacobian.

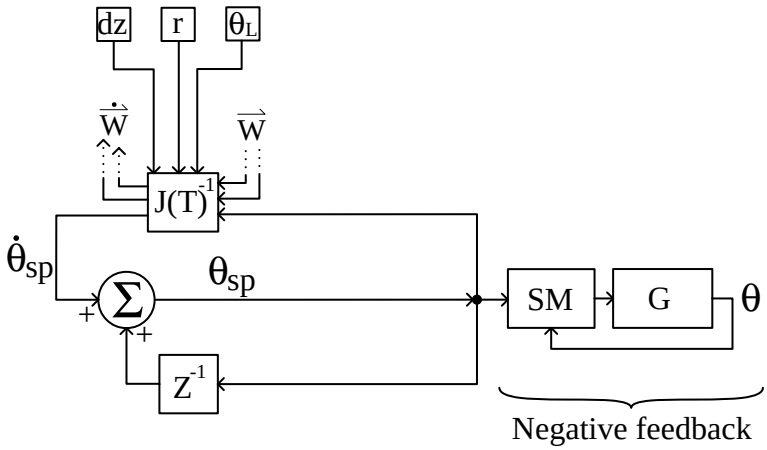


Figure 4.2: Open-loop velocity control diagram for one joint

This is a cascade controller for the position. The velocity is in open loop because there is no feedback from the real position. The position is in closed loop because the servomotor (SM) has its own feedback from the world (G); this is not shown in the equations. Going into the inverse Jacobian, is information from all joints under control ($\vec{W}$). The z^{-1} is a zero-order hold represented in Z-transform.

The controller as far as equation 4.2 represents the intended direction that would oppose all forces. This model now needs to be complemented with a positive feedback mechanism that would allow all joints to co-operate towards the same body motion.

4.2.1.2 Subordination: Positive Feedback Mechanism

Joints deviate from the intended set-point by certain amount, θ_d because joints are not stiff and are susceptible to external forces. In a positive feedback controller, the set-point is set to move according to θ_d regardless of the current leg intention, i.e., $\theta_{sp,t} \propto \theta_d$. The proposed solution is to include the positive feedback combined with the ideal $J(\vec{T})\vec{L}$ and the passive leg position $\vec{W}_d = [\alpha_d, \beta_d, \gamma_d]$ with a proportional subordination factor $\vec{S} = [s^\alpha, s^\beta, s^\gamma]$

$$\begin{vmatrix} \alpha_{M,t+1} \\ \beta_{M,t+1} \\ \gamma_{M,t+1} \end{vmatrix} = \begin{vmatrix} \alpha_{M,t} \\ \beta_{M,t} \\ \gamma_{M,t} \end{vmatrix} + \begin{vmatrix} \dot{\alpha}_{sp} \\ \dot{\beta}_{sp} \\ \dot{\gamma}_{sp} \end{vmatrix} \Delta t + \begin{vmatrix} s^\alpha \alpha_d \\ s^\beta \beta_d \\ s^\gamma \gamma_d \end{vmatrix}. \tag{4.3}$$

The subindex M indicates the cascade variable the position control is to maintain. As shown in equation 4.3, the new position to control is accumulated based on the previous position. Corresponding update to the previous control diagram is shown in Figure 4.3, note how the deviation is fed back modulated by the subordination parameter. It was mentioned above that equation 4.2 is in open loop, however, by introducing equation 4.3 the controller becomes closed loop because now it contains partial information of the real position. That is because the deviation $\vec{W}_d$ is the difference between $\vec{W}_M$ and $\vec{W}$, which is the real position.

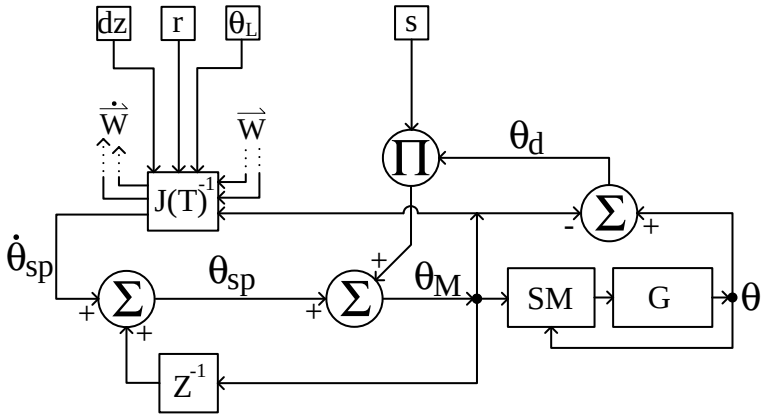


Figure 4.3: Closed-loop control diagram

This diagram expands Figure 4.2. Feedback comes from the deviation angle θ_d modulated by the subordination parameter s.

Without the term $\vec{S} = [s^{\alpha}, s^{\beta}, s^{\gamma}]$ on the right of 4.3 the controller would be in open loop; the same is true if $\vec{S}$ is zero. Therefore, the parameter $\vec{S}$ controls how much joints are subordinated to external forces. However, calculation of leg direction would be incorrect if one were to ignore the real segment position because the error accumulates continuously. The parameter $\vec{S}$ was named *subordination* for its role in influencing legs to follow external forces. Normally the term compliance is used for controllers that are restricted to certain positions, e.g., inserting a peg in a hole or polishing a surface. However, legs in the body do not have a hard constraint because all legs are controlled equally. Therefore, legs are not physically restricted to any path. To avoid confusions with standard terminology used in control theory it was decided to name it differently.

4.2.1.3 Individual Joint Activity

Results from section 3.7 suggest that it should be also possible for individual joints to change their activity independently. Furthermore, it was mentioned in section 2.4 that joints are not strictly coupled with each other. Hence, the following activation parameters were included:

$$\begin{vmatrix} \alpha_{M,t+1} \\ \beta_{M,t+1} \\ \gamma_{M,t+1} \end{vmatrix} = \begin{vmatrix} \alpha_{M,t} \\ \beta_{M,t} \\ \gamma_{M,t} \end{vmatrix} + \begin{vmatrix} \dot{\alpha}_{sp} \\ \dot{\beta}_{sp} \\ \dot{\gamma}_{sp} \end{vmatrix} \Delta t + \begin{vmatrix} s^{\alpha}\alpha_d \\ s^{\beta}\beta_d \\ s^{\gamma}\gamma_d \end{vmatrix} + \begin{vmatrix} \Phi\kappa^{\alpha} \\ \Phi\kappa^{\beta} \\ \Phi\kappa^{\gamma} \end{vmatrix}. \tag{4.4}$$

The function Φ is the activity of the leg and is modulated by κ. In particular we suggest that κ^{α} should be non-zero only for the metathorax and κ^{γ} should be non-zero for the mesothorax in agreement with the model proposed in section 3.7. Thus, for each motor joint the following equation is used,

$$\theta_{M,t+1} = \theta_{M,t} + \dot{\theta}_{sp,t}\Delta t + s\,\theta_{d,t} + \Phi\kappa \tag{4.5}$$

The complete control diagram is shown in Figure 4.4. The six square blocks at the top represent the control variables, the three on the left, dz, r and θ_L are the intended tarsus velocity. These in turn depend on signals from the brain. Note that in the implementation below the leg direction $\theta_{\vec{L}}$ and Φ will in practice be set either to zero or to the direction of the visual target ϕ. The input dz varies to control the local height and the variable r is held constant.

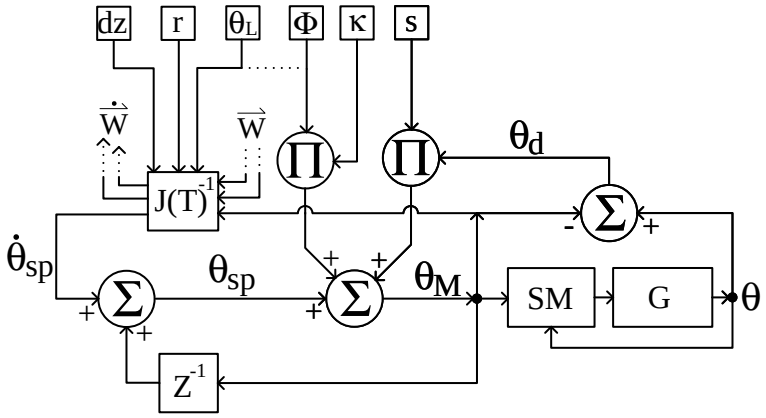


Figure 4.4: Complete control diagram

This diagram extends Figure 4.3. Additional activation can be introduced by the parameter Φ modulated by the parameter κ. For the control of turning function $\theta_{\vec{L}}$ and Φ are made equal to ϕ for the front legs.

4.2.2 Subordination Study: 2D Limb Simulation

To understand how this controller works and what the concepts behind it are, a simpler system that contains similar elements is introduced. The system, a two-degree of freedom limb, is to move its tip $tp(x,y)$ along a straight line. It is controlled by two servomotors and there are two external forces opposing its movement: gravity and a predefined force at the tip. Servomotors are connected to segments as shown in Figure 4.1, i.e, it has non-stiff joints. This scenario was numerically programmed using Matlab®, the mass and shape of the elements are explained in appendix B.1. The initial position for the limb is at α = 140 and β = 20, which is equivalent to $tp \approx [-0.3, 0.18]^T$ meters. The joint angular velocity is calculated with the following equation.

$$\begin{vmatrix} \dot{\alpha} \\ \dot{\beta} \end{vmatrix} = J_{tp}^{-1} \begin{vmatrix} \dot{x} \\ \dot{y} \end{vmatrix} + \begin{vmatrix} s^{\alpha}\alpha_d \\ s^{\beta}\beta_d \end{vmatrix} \tag{4.6}$$

This a good example for analysing leg dynamics as it only has one hard constraint; the ground. Similarly, legs are strongly attached to the ground and body, but the position of the latter is variable. Furthermore, variability is proportional to the flexibility and number of joints, i.e., the more joints and the more flexible these are the less constrained its motion. The external force on the tip represents the body contact, due to

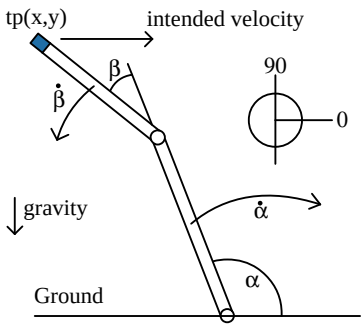


Figure 4.5: 2 degrees of freedom limb used for description of the leg controller

The 2 DOF limb is fixed only at one point (base) and the tip is controlled by two motors. The tip was always to move horizontally to the right; arrows indicate the initial angular velocity for this to happen.

other legs or its own weight.

The limb was given one second to get to the starting position in order to avoid force peaks at the beginning due to variable initializations. The position was held for one second and thereafter it was instructed to move horizontally to the right, $\{\dot{x} > 0, \dot{y} = 0\}$.

Initially, the limb behaviour influenced only by gravity is studied, with subordination values s^{α} and s^{β} varying from 0 to 0.34.

Figure 4.5 shows with arrows the initial angular velocity the joints need to follow for the tip to go along a horizontal trajectory. Angular velocity of the α joint is always negative, moving the limb to the right. β is positive for the first two thirds of the trajectory and then it must change direction. Gravity aids movement of the α on positions below 90 degrees and β on positions where $\alpha + \beta > 90$. The former is reached after 0.2 seconds, but the latter condition is maintained for the whole movement.

Figure 4.6 shows results after varying subordination parameters in both joints. Deviations from the expected horizontal trajectory are non-linear and mostly unpredictable. Depending on values given to both subordination joints deviations are directed differently. Note that it is the combined subordination that determines the final direction. The β joint has a greater effect on the height of the tip and the α has a greater effect on the lateral displacement. Nonetheless, there was more vertical variation when $s^{\beta} = 0$ than when $s^{\beta} > 0$ because gravity and the initial intended angular ve-

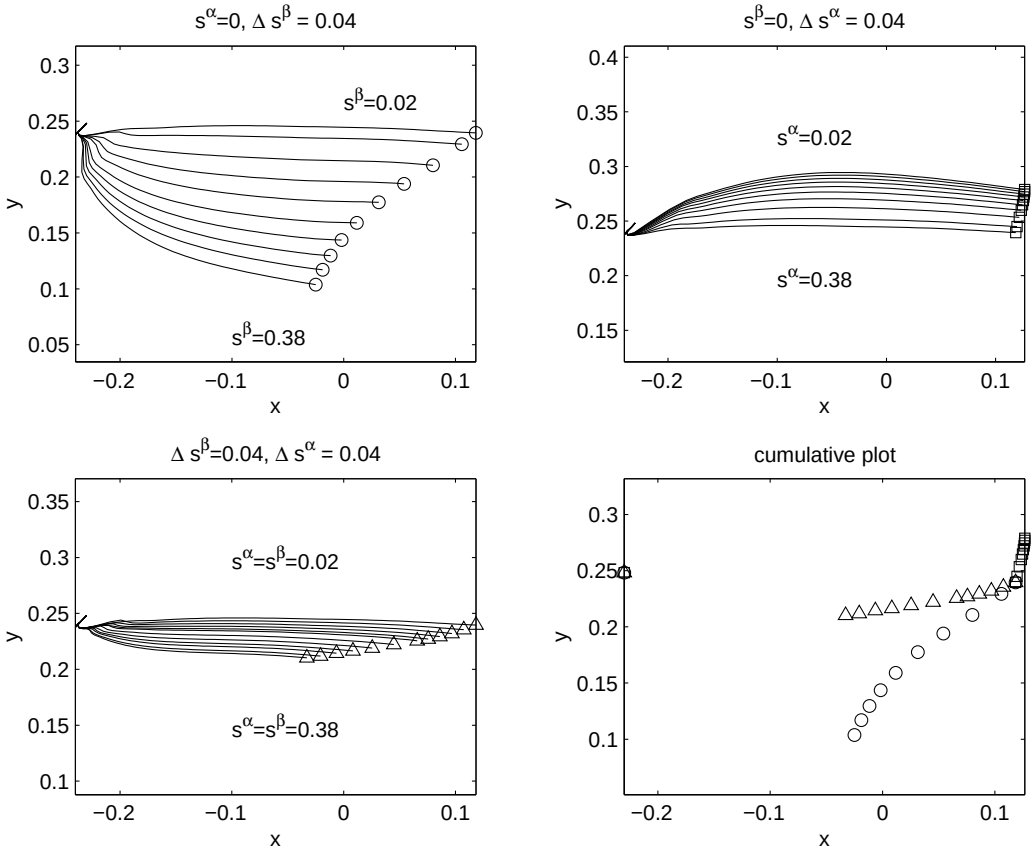


Figure 4.6: Limb moving free of external forces with internal parameter variations

The subordination parameters s_α and s_β were varied, gravity was constant $g = -9.8\hat{y}$. Top left: $s_\beta = 0$ and $\Delta s_\alpha = 0.04$ starting at $s_\alpha = 0.02$. Top right: $s_\alpha = 0$ and $\Delta s_\beta = 0.04$ starting at $s_\beta = 0.02$. Bottom left: $\Delta s_\alpha = \Delta s_\beta = 0.04$ starting at $s_\alpha = s_\beta = 0.02$. Bottom right cumulative plot showing only end points.

locity supported each other. Small trajectory variation when $s^{\alpha} > 0$ happened because geometrically the gravity affected that joint less. It is important to note that although direction is calculated with inter-joint information, subordination is local. The final direction will depend on the relative difference between and s^{α} and s^{β}, not only on their absolute value. For instance, the large vertical variation when $s^{\beta} = 0$ was because the β joint was intended to move initially in the direction of gravity, $\dot{\beta} > 0$. Nonetheless, the α joint inefficiency to move right resulted in the tip following only $\dot{\beta} > 0$ motion. Relative to the motion specified in the task, the figure on the bottom left corner is the least deviated. Nonetheless, it is important to bear in mind that the objective is not to control direction or speed per se, but to accomplish the three points mentioned at the beginning of this chapter. Thus, all trajectories shown in Figure 4.6 have some useful application depending upon specific tasks.

4.2.2.1 External Forces

It is important to understand how the controller behaves not only for continuous forces like gravity, but also with unexpected forces. After the limb was moving for one second a horizontal F_x or a vertical F_y force was applied to the tip for one second. Subordination was kept constant, both joints with a value of $s^{\alpha} = s^{\beta} = 0.06$. Figure 4.7 shows results for forces in both directions. Forces lasted for one second after the force was applied, at that point the limb's geometry prevented it from moving any further.

Direction changes as the external force increases, for the extreme case reported it almost reaches a state where $\dot{y} < 0$ and $\dot{x} \approx 0$. However, after approximately 20ms the system stabilizes again to move horizontally, $[\dot{y} \approx 0, \dot{x} > 0]$. Negative horizontal forces tend to decrease the speed of the tip, but after the system stabilizes, the direction is again as intended. On the other hand, vertical forces do not reduce speed; they mostly affect the vertical velocity of the tip. In both situations, the limb is subordinating itself to the external force (intended direction). Nonetheless, even after abrupt changes in direction and impulse, the tip direction still tries to go in the intended direction.

4.2.2.2 Force Pulse

One indicator that a variable is not under position control is when the external force is transiently applied but the intended position cannot be reached. All controllers correcting position have a theoretical steady state error of zero once the external force is removed. Results are now presented on how the system responds when the external

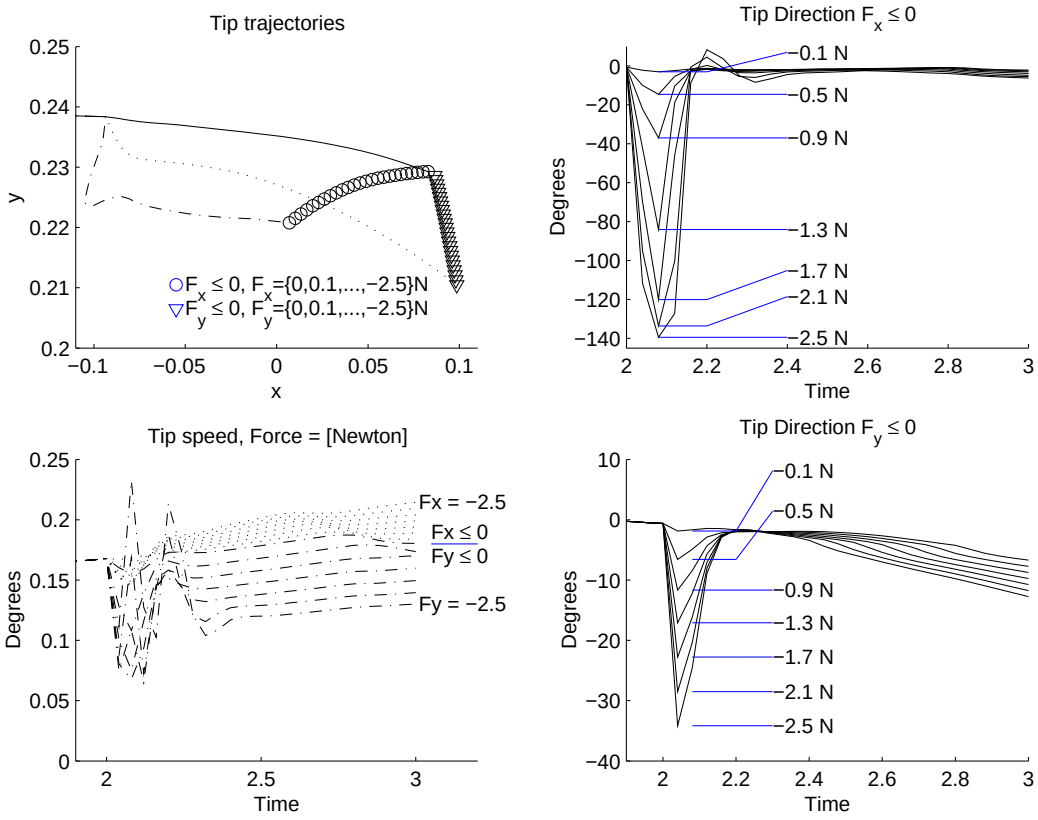


Figure 4.7: Limb moving to the right with external step force

The external force started at t=2 sec. Top left: Different end points in response to different force strengths. Circles: forces in $\hat{x}$. Triangles: forces in $\hat{y}$. Full trajectories corresponding to the response to the force F_x=-2.5N, F_y =-2.5N and F = 0 are shown. Right: Directions of the tip responding to lateral forces (Top) and vertical forces (Bottom). Left bottom shows the speed profile of the tip in response to different forces.

force is introduced after one second but then removed. The tip of the limb for this experiment was to maintain a null velocity [$\dot{x} = 0$, $\dot{y} = 0$]. Gravity was removed and a vertical external force pulse of -1 Newton was applied for $\Delta t = \{100, 200, ..., 900\}$ ms. Two set of subordination values were used, $\{s^{\alpha} = 0.1, s^{\beta} = 0.2\}$ and $\{s^{\alpha} = 0.2, s^{\beta} = 0.3\}$.

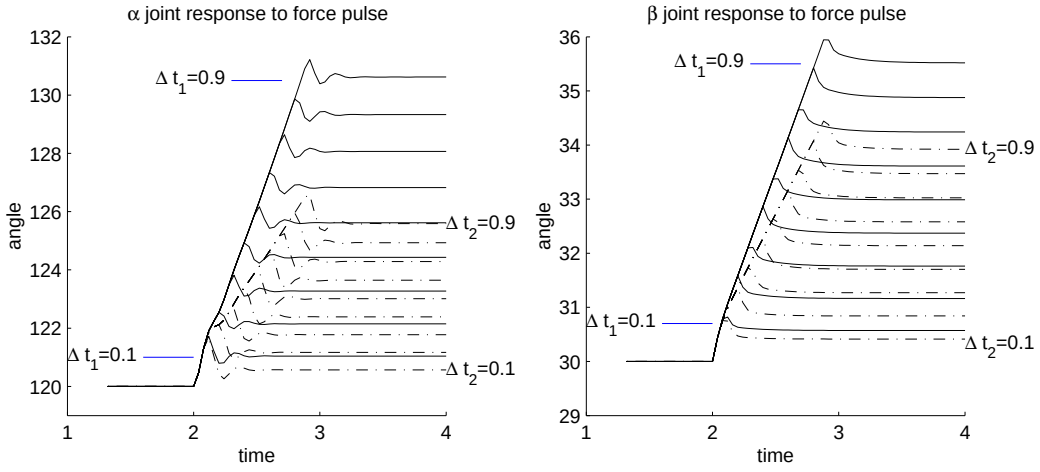


Figure 4.8: Joint response to a force pulse for two sets of subordination values

Continuous lines and Δt_1 refers to subordination $\{s^{\alpha} = 0.2, s^{\beta} = 0.3\}$; Discontinuous lines and Δt_2 refers to $\{s^{\alpha} = 0.1, s^{\beta} = 0.2\}$. To the left the α joint and to the right the β joint.

Figure 4.8 shows results for the α joint on the left and results for the β joint on the right. Even for short pulses, it is evident that the original position is never restored for either joint. In addition, whilst the force is active the angular distance to the origin increases linearly. This suggest that the controller is velocity-based, because a position controller would eventually move to a resting position proportional to the force. However, as seen in Figure 4.8, both angles move at a steady rate as long as the force is active.

4.2.3 Generic Control Model

The model described so far represents the mechanical arrangement shown in Figure 4.1. However, the controller proposed can be implemented on mechanisms that lack the spring box as well. For this, it is only necessary to scale the subordination and

the activation in equation 4.4. This is now tested by removing the spring chassis and connecting the segment directly to the motor. It will be demonstrated that to obtain similar dynamic responses, it is only necessary to adjust the subordination parameter accordingly.

Assume a controlled motor is set to control position θ_M. Because the controller is not perfect, it would get to position θ_S. Furthermore, because of mechanical constraints, the segment would not be at angle θ_S, but at θ. The deviation used in equation 4.3 uses the real position and the position sent to the motor, the position of the spring chassis is not used (Figure 4.9). The springs increase the error and this in turn aids the reading of the angular difference. However, this increase is theoretically proportional to the spring constant and it should be possible to compensate it with the subordination parameter. This opens the possibility to a variety of other possible implementations.

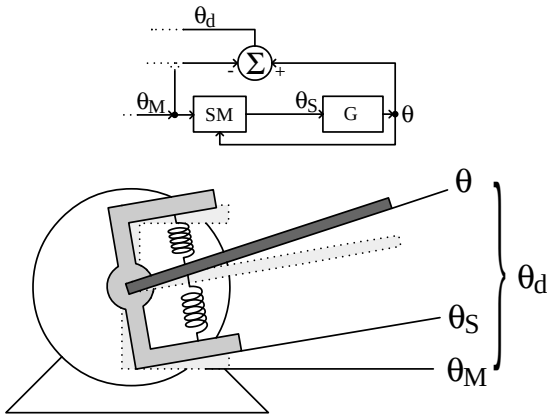


Figure 4.9: Deviations at the joint

On the top, the section of the control diagram focuses on the physical implementation. There are two deviations: that between the motor command θ_M and the actual motor chassis θ_S; and that between the chassis and the actual segment position θ. However, the controller only uses $\theta_d = \theta - \theta_M$.

If a PD controller is chosen for the motor, the theoretical angular increase towards the set point would not reach the target, $\varepsilon = \theta_S - \theta_M$. The torque would be proportional to ε and so the system is equivalent to having a spring between the target and the motor[2], the spring constant equivalent is K_{MS}. A realistic servomotor-segment system could be represented by three segments attached by two springs of different force

[2]The derivative has no effect on the steady state error.

constants. The first one is the set motor position, the middle is the motor chassis position, and the last is the actual segment. Spring constants are defined as K_{MS} for the one between target and motor and K_{SR} for the one between motor and segment. By defining these spring constant all joint mechanisms can be described. By measuring the deformation of these springs, it is possible to calculate the force each of these are producing.

Intentionally, the model described before assumes there are springs between the motor chassis and the segment. This allows to measure deformation directly without using strain sensors. However, the controller proposed before does not use the real position of the motor, just the target and the real segment position. The subordination is dimensionless and so far, values are chosen heuristically. Therefore, if the spring system is not used, resulting in a stronger K_{MS}, it would only be necessary to increase the subordination s. Therefore, any system that has a difference between the target position and an approximate position of the real angular position can be controlled with the previous equation. If the deviation is too small, it is still possible to use a stress gauge. Alternatively, it is possible to use a weaker PD controller.

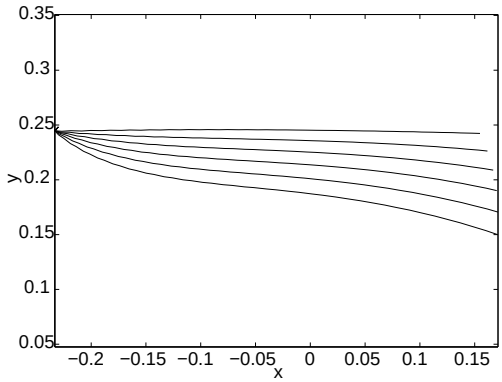


Figure 4.10: Limb with no spring between motor and segment

The integral component of the motor control was removed and the deviation between set-point and segment was used for θ_d. Results are similar to Figure 4.6.

In the previous experiment, the position of the segment was used to calculate the deviation θ_d. Assuming the controller was perfect, the deviation was only attributed to the spring box. However, the springs just increased the deviation error. If the segment-motor PID controller does not have the integral term PD, it would behave more like a spring between θ_M and θ_S. Therefore, it is possible to use the motor controller

itself to emulate the spring box. Figure 4.10 shows the response of the limb when the integral term is removed from the servo controller; the spring box is removed; and subordination was increased to $s^{\alpha} = s^{\beta} = 0.5$. The response of the system can be similar to that using the spring box, cf. Figure 4.6.

However, mechanisms shown in Figure 4.1 can be easily implemented in the robot simulation. Furthermore, it increases the stability of the simulation and thus the implementation below in fact includes a spring box. This discussion suggests a range of mechanical implementations would be compatible with the leg controller scheme.

4.2.4 Summary

Experimental results demonstrate that legs always react with a negative feedback mechanism to external forces. Experimental results in chapter 3 also show that legs react with a negative feedback in reaction to inter-leg forces. However, tarsi do not slip and therefore joints move synchronously. The model proposed herein, has a core negative feedback controller influenced by a positive feedback modulated via a subordination variable $\vec{S}$. Individual joint activation can also be controlled, allowing to influence the direction of stance by a simple mechanism.

The behaviour of this model, was illustrated by implementing a 2 DOF limb controlled with non-stiff servomotors. It was also demonstrated that the limb controls its direction and simultaneously allows itself to be influenced by an external force. However, the internal intention is never affected, thus the trajectory is not compromised. The controller can behave like a position or velocity controller depending on the values given to the subordination variables. For non-zero values, the controller resembles more a velocity controller, thus supporting results found for the stick insect. Additionally, the position controller could be complemented with an integral compensation mechanism, which would increase the resemblance to the biological target.

The controller was implemented in a non-stiff joint that operated by interfacing motor and segment via a series of springs. However, although this improves controllability, the controller is in theory applicable to all situations where there is a deviation between desired position and real position.

4.3 Robot Simulation

The controller was next tested in a full hexapod system. The robot simulation where hypotheses were tested is basically divided into two integrated simulations: The Open Dynamics Engine (ODE) and the robot simulation. The physics world simulation calculates the motion of rigid bodies and their interaction (Fig. 4.11). The world simulation used is a free-licensed software registered by Russell Smith. These libraries were not modified for experiments presented herein, these are described in more detail in appendix B.2.

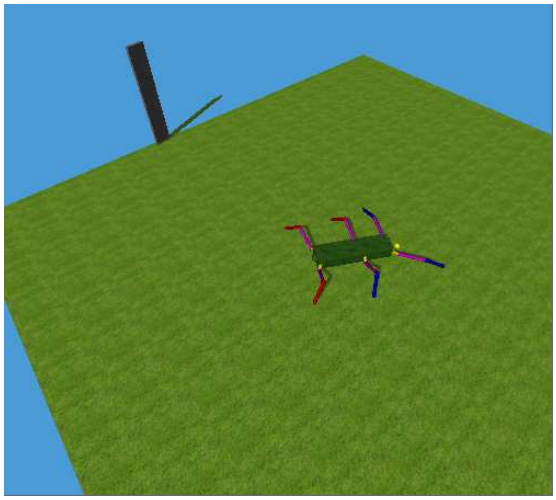


Figure 4.11: Robotic simulation environment.

The robot simulation uses objects available from the ODE libraries to construct an insect-like robot. It was programmed to match as much as possible mechanical properties of real robots, for instance, the way motors operate and the sensors available for the walking controller. However, just as with the body dynamic simulation, approximations could not be avoided.

The ODE simulation runs by numerically estimating expectation of body states by small increments in time. Because error accumulates, it is recommended to have a small simulation time step. For most simulations, this was set between 0.1 and 1 millisecond. However, the walking controller should comply with delays normally found with available electronics. Therefore, the robot simulation, sensors and motors, run at a different frequency, normally between 5 and 20 ms. This time is perhaps still optimistic, but it is important to have both run times independently.

4.3.1 Morphology

The robot simulation was designed using the basic shapes available; these were the hexahedron, sphere and cylinder. Rigid bodies were brought together by hinge-joints at different positions. The size and relative weight of segments correspond to that of a robot and not an insect. Dimensions of main body segments are given below,

Segment	Dimensions (m)	Weight (kg)
Body	$0.32 \times 0.48 \times 1.82$	22.4
Coxa	0.08, 0.08 ⌀	0.96
Femur	0.60, 0.08 ⌀	3.20
Tibia	0.56, 0.08 ⌀	3.20

Table 4.1: Body segment dimensions and weight

The prothorax and metathorax pair of legs were positioned at the extremes of the body, i.e. at positions 0 and 1.82 along the body; the mesothorax was located at 0.6 from the prothorax, i.e $0.6 \times 1.82 = 1.09$ along the body. Each leg is specified by the three segments described before, and five hinge joints. The first two hinge-joints are fixed to a given angle and orient the coxa away from the body. These are ψ'' and ψ', where the hinge-joint ψ'' was set to zero for most experiments, as described in section 2.2. The other three joints actively move the leg and are generally known as α, β and γ joints.

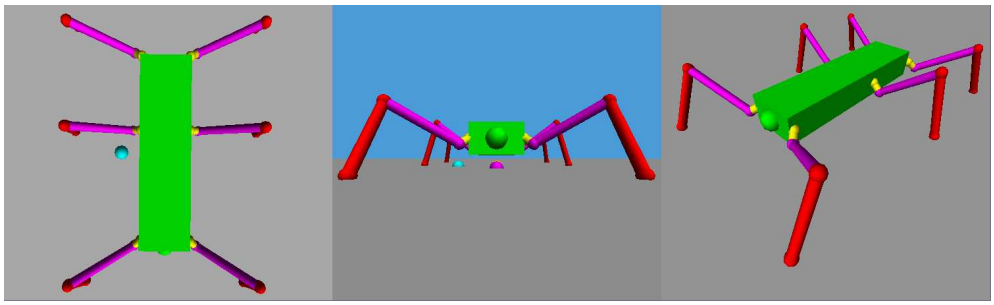


Figure 4.12: Robotic simulation, initial position.

Each of these joints was programmed to be controlled by a servomotor and each can move only within certain angular range, depending on the thoracic position. They are located at the following table shows the parameters used for most experiments,

Prothorax	Mesothorax	Metathorax
$-80 < \alpha < 30$	$-40 < \alpha < 40$	$-40 < \alpha < 40$
$-30 < \beta < 80$	$-30 < \beta < 80$	$-30 < \beta < 80$
$-120 < \gamma < -20$	$-160 < \gamma < -20$	$-160 < \gamma < -20$

Table 4.2: Range of angular movement in joints.

The initial position was predefined to the same position, the values given to the different angles are shown in the following table and shown in Figure 4.12

Joint	prothorax	mesothorax	metathorax
α	-40	0	30
β	57	75	67
γ	-100	-110	-100

Table 4.3: Initial joint position.

These parameters specifying the robot morphology could not be changed after compilation. However, it is straightforward to change robot morphology. The robot simulation at various sizes and weights is tested in due course. Figure 4.12 shows the initial position described in the table above.

4.3.2 Motors

The robot controls its movements by changing the angle of the 18 leg joints. Each of these joints can be controlled in two different ways, by specifying torque or setting angle position. Although the former is more suitable to model biological systems it is not yet easily implemented on a robot. The most common motor implementation for motion control is the use of servomotors, of these, the most common control angular position. The position control is usually implemented by the servomotor, thus it is not necessary to calibrate or design this controller. Maximum torque ratings vary from less than 3 Kg cm[3] to a couple of tens on heavy duty motors, or above one hundred on industrial models.

The ODE libraries implement their own motors for controlling joints. However, these add further errors to the approximation of motion and increase the computing

[3]Produces one kilogram of tangential force at one centimetre off the axis.

time. Instead, it is possible to bind joints in the simulation by having the same minimum and maximum angular position and vary this value to emulate the servo behaviour. It is then possible to emulate spring and damper values to joints[4] that would allow joints to be moved by external forces independently of the internal position control. The resulting mechanism is no different than that shown in Figure 4.1 and implemented in section 4.2.2. The resulting deviation is illustrated in Figure 4.13, where the hind leg is drawn with lines. The lower version of the leg (purple) represents the position set and the higher version (red) is the real position. Note that the position set goes beyond the ground because the leg is controlling height; thus, the leg is pushing down.

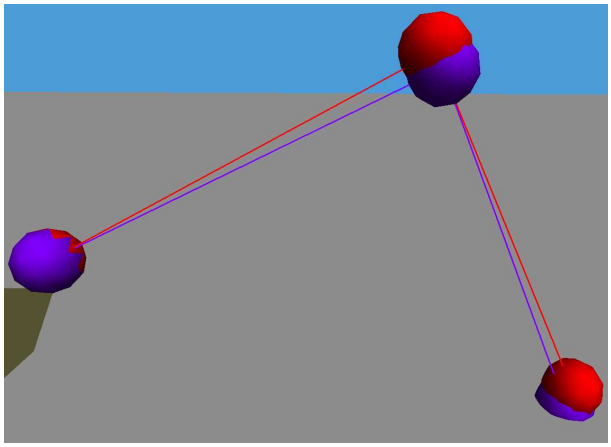


Figure 4.13: Visual representation of deviations in the hexapod leg

Left: Deviation at the CT and FT joint.

Therefore, motors implemented in the simulation are position controlled by setting the joint bounding conditions to equal the desired position. This in turn would depend on external forces because the parameters set for ERP and CFM would emulate the motor configuration shown in Figure 4.1.

4.3.3 Sensors

All simulation variables could potentially be used as sensory input. However, sensors were only used when there was sufficient support to believe the insect had access to

[4]These can be set by adjusting the ERP and CFM in the ODE libraries. Appendix B.2 explains these parameters in detail.

them as well. For instance, the simulation has available global position, velocity and acceleration of the body. However, it is not known if that information is available to the insect, and if accessible, it would have to be calculated indirectly based on leg position.

However, even by restricting the type of sensors to those possibly available to the insect, the different readings available are numerous. Some sensors that are known to be available to the insect were used only sporadically because their role in walking is not yet fully understood. For instance, load sensors on the BC joint are known to influence walk stepping. Additionally, the animal has information about acceleration and vibration at joints, which can convey information about surface properties. However, as mentioned before, it was not possible within the time given to include all current knowledge.

Because of programming structuring, internal information that was obtained after sensory processing was also labelled as sensory information. For instance, the distance legs travelled during stance is not obtained directly, but estimated after leg motion. These parameters could have been estimated directly, nonetheless, direct information from the simulation was avoided.

The only sensors a robot would require to implement the controller proposed herein are the angular position and velocity at the joints. Additionally, load signals at the tibia were used. The latter sensor was only used for terminating swing, hence, it could be replaced by load sensors at a different location.

Unfortunately, vision signals for targeting were not properly implemented because of time constraints, i.e., visual processing and eye properties were not implemented. The angle to the target was directly calculated relative to the front of the body.

4.4 Results

This section examines how a hexapod system as described in the previous section responds in various situations. Leg coordination implemented in the robot simulation is explained in more detail in section 5.1. However, when used only for forward walking the response is as described in (Cruse, 1990). The robot simulation always walks slightly differently because the leg coordination is probabilistic. That in turn affects the likelihood of legs moving optimally, which changes the overall walking. If not stated otherwise, the simulation was always run three times and results were averaged accordingly for a given set of parameters. Only when results were consistently similar are representative trajectories shown. The height was controlled with the CT joint (β),

and the BC (α) and FT (γ) joints with equation 4.3.

Tests on the robot are divided into three sections. First, forward walking is studied on a flat surface for a set of different subordination values. Resulting deviation at each joint is identified, and how this affects the tarsi direction relative to the body. This is presented in section 4.4.1. In a second experiment, the body trajectory is affected by two different forces, one in the direction of travel and the other perpendicular to it. This situation is not commonly found on flat surfaces. However, a more realistic approach with similar forces results when the robot walks on a rotated surface. Therefore, the resulting body trajectory was analysed when the surface is tilted and when the surface is rolled in section 4.4.2. The controller does not use body position to control its motion, i.e., only individual legs are controlled. The last experiment in section 4.4.3 studies the relationship between leg speed and body speed.

Results with the hexapod robot simulation shown in this chapter were performed after solving the control of turning in the next chapter. No modifications were made for this chapter other than those that required parameter variation. Therefore, along with demonstrating how the controller was implemented, this chapter aims to demonstrate that forward walking can be assumed to be a simplification of turning. Note however, that the opposite cannot be easily achieved, i.e., solving for forward walking does not necessarily solve for turning.

4.4.1 Subordination Variation

Each leg was set to walk at an average speed of 0.1 m/s and at zero degrees, i.e., walking forward. Subordination was varied proportionally on the α and γ joint. The subordination $\vec{S} = [s^{\alpha}, s^{\gamma}]$, was given the following 7 pair of values s^{α} = [0.02, 0.04, ..., 0.14] and s^{γ} = [0.03, 0.06, ..., 0.21]. The α joint was given a lower value because it is mainly responsible for driving the body forward. For the lowest values $\vec{S}_1$, the pattern followed by the angles during stance was close to the optimum, i.e., $s^{\alpha} = 0.02$ and $s^{\gamma} = 0.03$.

The resulting deviation in individual angles as subordination increased was largest for α. That was expected, as it is the joint with more opposing force relative to its subordination value. The resulting deviation of the β joint was similar to that of γ because it controlled changes in height caused by the γ joint. Results are shown in Figure 4.14, columns are the leg joint angle resulting deviation and the rows are the three thoraces. It is not clear why the front legs produced larger deviations. Note

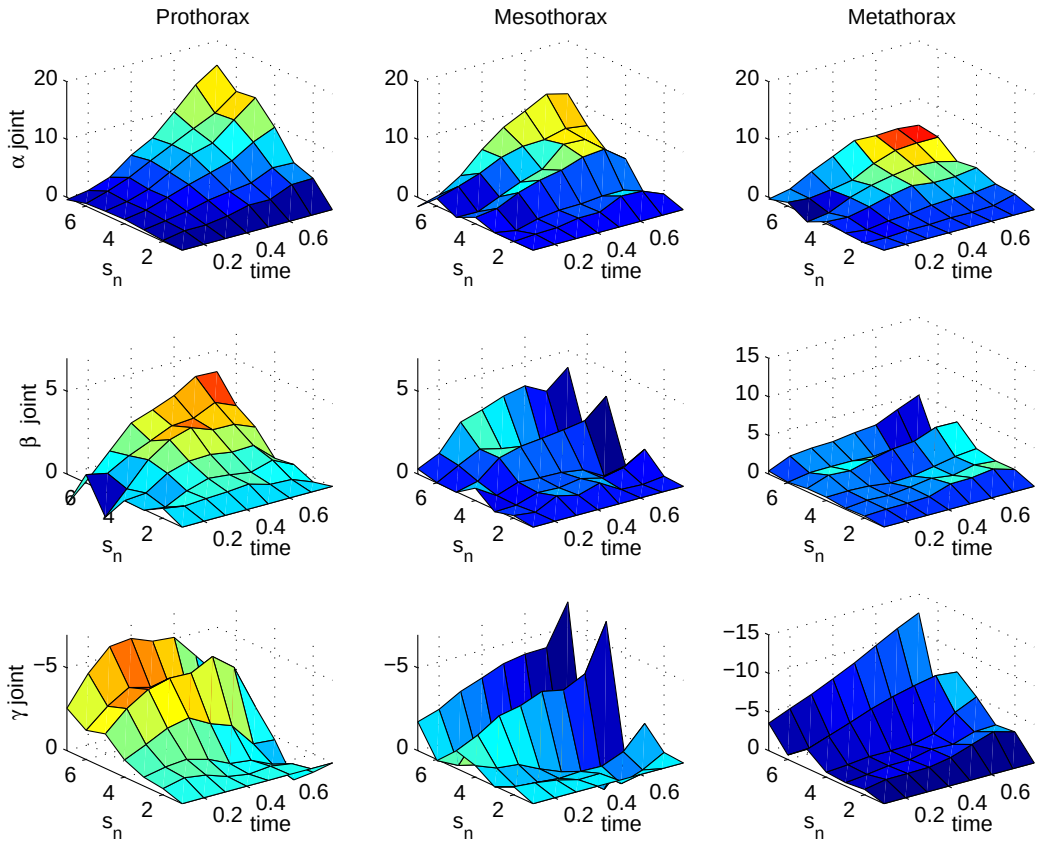


Figure 4.14: Angle deviations in joints

Angle deviation during stance is plotted against time from the AEP and 7 subordination parameter sets of values s_n, where $n = \{0..7\}$. Statistics for pairs of legs were merged. Column results for the three thoraces and rows for each of the leg angles. The β joint compensates for much of the γ deviation because is controlling height. Note that the γ joint has negative values and the different scales.

that the set label of subordination values is shown and not the actual subordination value for each joint. It is possible that because the centre of mass is towards the rear, resulting body torques affects them more. Nonetheless, no quantitative explanation was found. However, it might be just the resulting geometry. The robot could walk for all sets of values, however, the direction was deviated when using the last two sets of subordination values.

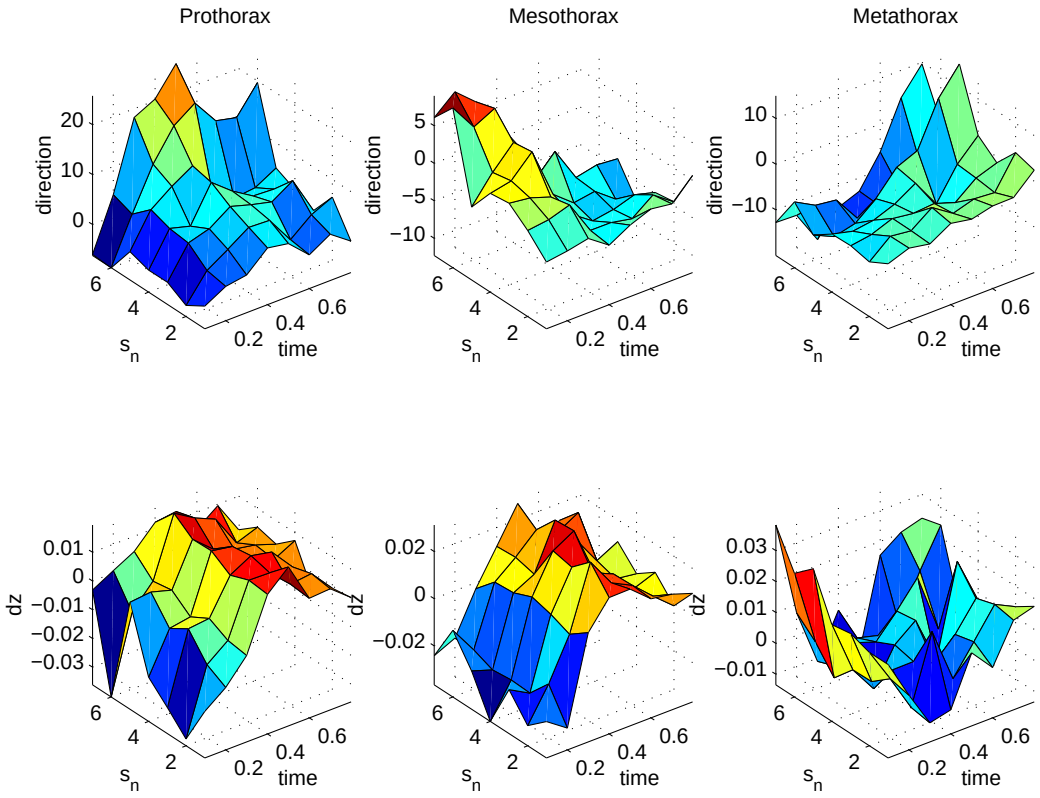


Figure 4.15: Velocity deviation in legs (polar coordinates)

Leg tarsi direction parallel to the body during stance for 7 different subordination parameters. As subordination increases, leg direction reaches as much as 20 degrees' deviation on the prothorax. Time is from the moment legs were at the AEP.

Leg trajectory directions are shown in Figure 4.15, the top row is the direction and the bottom row shows vertical movement. The first set of subordination values $s_{n=0}$, show direction deviations very close to zero degrees, however, during this time height variation is larger. For the last set of subordination values, legs were moving at a significantly deviated average direction, as high as 20 degrees. The rotation point was somewhere around middle legs because these legs show less error. Note how

variations in height dz, do not vary as subordination increases because the control of height is always correcting for this. Additionally, note that initially dz varies the most, this is because the ground is never found at the height the local leg tries to maintain. Remember that the leg stops swinging when the leg starts supporting the body and not at its ideal local height.

4.4.2 Surface pitch & roll

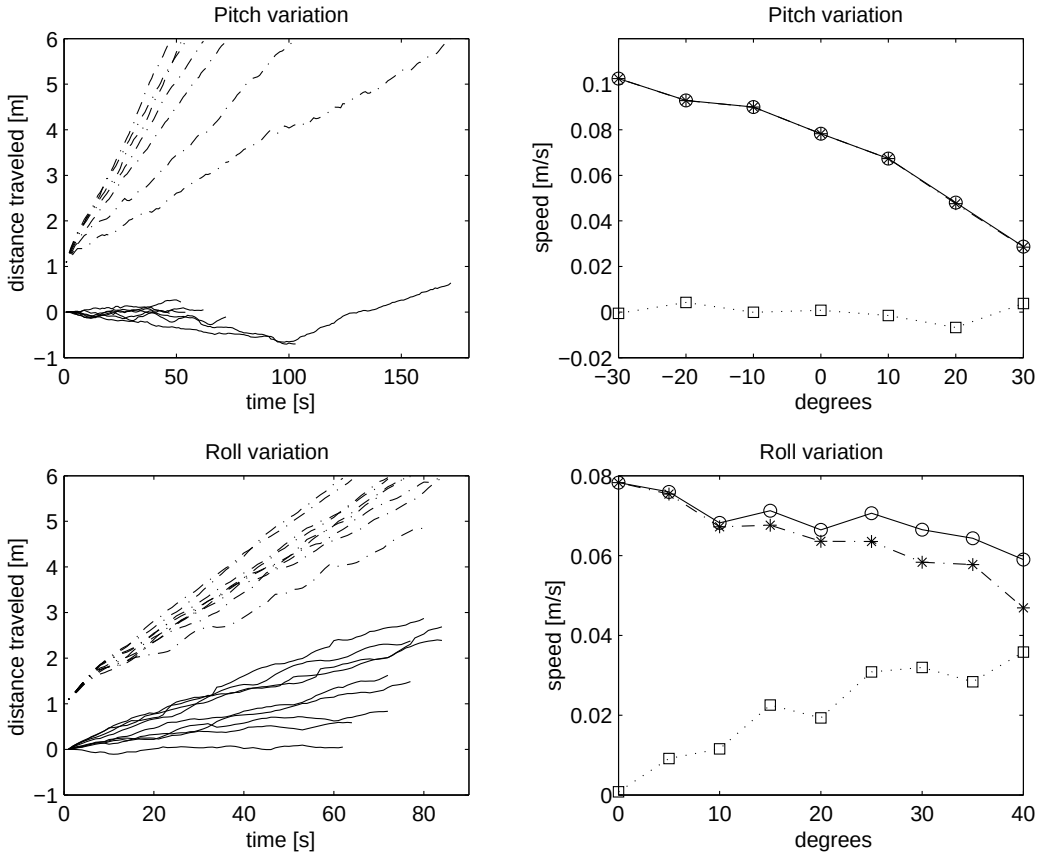


Figure 4.16: Robot simulation response to the surface being tilted and rolled

Simulated robot moving on a tilted surface; results are for the prothorax. On the two top figures, the simulation ascends or descends; force coincides with the direction of walking. On the two bottom figures, the surface is rotated to one side; force pushes the body to one side. Body speed (○) is shown decomposed into forward speed (∗) and lateral speed (□). On distance vs. time plots, decomposed velocities are shown with non-continuous (forward) and continuous (lateral) lines.

The system was tested with different surface orientations in the direction of travel (pitch) and perpendicular to initial heading (roll). The robot faces in the direction of $\hat{x}$ and lateral drifting is in direction $\hat{y}$. Legs try to move parallel to the body, as there is no other sensory information. Speed and subordination values were kept constant at $r_{\vec{L}} = 0.1\,ms^{-1}$ and $\vec{S}$ = $[0.06, 0.09]$ for all experiments. Results are summarised in Figure 4.16.

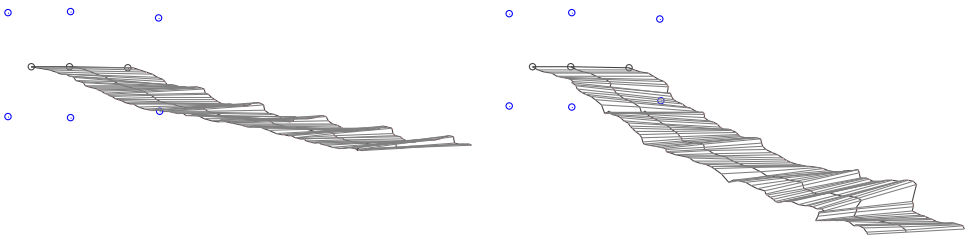


Figure 4.17: Representative body trajectories for surface roll angles

Top view of trajectories, only initial leg positions are shown with circles. Left: θ = 10. Right: θ = 40.

The top figures show results varying the pitch angle and the two bottom figures show results when the surface was rolled. Forward walking speed changes almost linearly with the pitch angle of the surface. Trajectories show only small lateral deviations. Climbing slopes bigger than 30 degrees had a large failure rate and the speed was almost zero. Pitch angles below -30 degrees make it very unreliable. Nonetheless, at these extreme pitch angles most failures were caused by tarsi slippage.

Variations in roll angle show a positive result because the robot is compliant with directions not under control, i.e., lateral direction. Roll angles larger than 40 degrees resulted in the robot tumbling down. Furthermore, the total average body speed when walking on a rolled surface did not changed much, at 40 degrees the speed 0.06 m/s, is similar to the climbing speed at 10 degrees. Walking was mostly parallel to the intended direction as shown in Figure 4.17 even though is moving laterally. The lateral deviation could have been prevented by increasing activity in the FT joint. However, this is more likely decided at a higher level of control. This experiment shows how the system responds when the external force coincides with the movement under control and when the force does not interfere directly. It was not the intention of this experiment to try to compensate for the lateral drifting. Nonetheless, it is important to

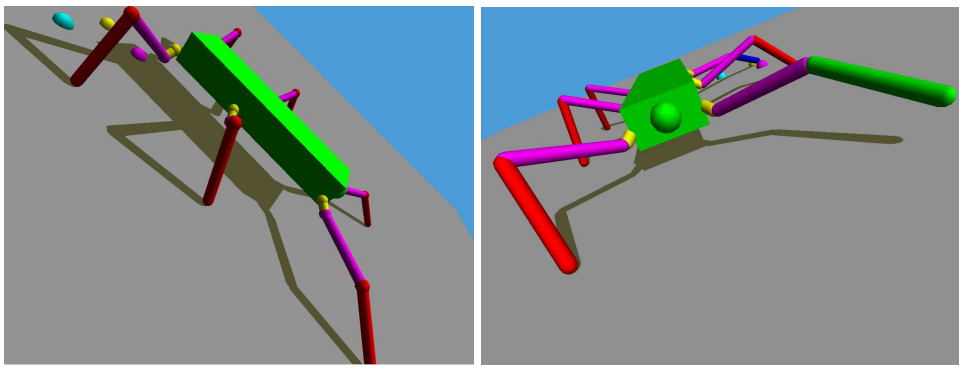


Figure 4.18: Snapshots of the robot simulation when the surface angle is altered

Left: Holding zero velocity on slope. Right: Walking with horizontal force; legs on the right are closer to the body.

note that simple solutions could be included to reduce this effect. Such as increasing activity of FT joints by detecting visual slip or body pitch.

Figure 4.18 shows two snapshots of the simulation while walking on a tilted surface. Leg geometries change without explicit control. On the left, the robot is trying to maintain the initial body position (null speed). The front legs eventually rest extended and positioned to the front, in what seems to be a qualitative optimal orientation based on the direction of pulling and the geometry of the legs. On the right, leg geometry between contralateral legs is very different, however, forward walking is still feasible as shown in Figure 4.17. Experiments by Diederich et al. (2002) on stick insect walking along inclined surfaces report a leg geometry like that shown in Figure 4.18. Additionally, they also report changes in AEP and PEP that would result in the body drifting towards the direction of gravity. Unfortunately, no body trajectory is reported and further comparisons were not possible. Similarly related is research by Dean (1991), where the speed of the stick insect is known to decrease or increase when a force is applied in the opposite or the same direction of travel respectively. As the surface angle increases, most instabilities were not related to the controller, i.e., some could have happened with the static robot. For instance, roll angles above 40 failed before the first step was completed. It is important to note that the simulation was not programmed to cope with variation in surface rotation, nonetheless, the robot performs very well.

4.4.3 Speed variation

The robot simulation was set to walk at different speeds to study how the system responds to its own dynamics. For instance, when standing still, front legs tend to create forces going backwards whereas hind legs create forces towards the front because of the body geometry (Cruse, 1976b). Results shown in Figure 4.19(a) indicate a linear relation between individual leg speed and body speed. The leg speed shown is the one sent to the leg controller and not the real leg speed. Previous experiments indicate that the relation between subordination and tarsus speed is non-linear and it is different depending on which thoracic segment legs are on. However, it is important to note that for a given leg subordination the speed of the body can be easily controlled because it is proportional to the speed sent to the leg.

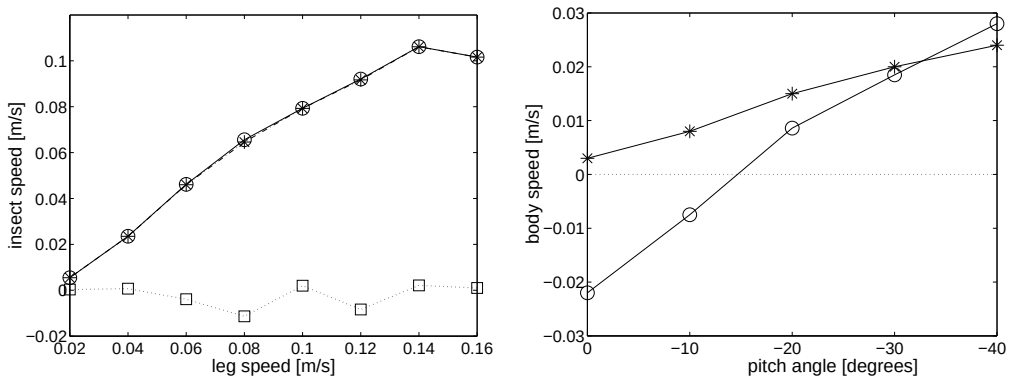


Figure 4.19: Body speed dependency on legs speeds

Left: Body speed relative to leg intended speed. Body speed (○) is decomposed in lateral speed (□) and forward speed (∗). Right: Resulting speed when leg intended speed is zero but the robot is on a slope and being pulled forward by gravity. Experiment without legs being able to swing are shown with (∗) and with legs being able to swing are shown with (○). Refer to text for explanation on the difference in resulting body speed.

4.4.3.1 Zero Speed

In experimental results with the 2D limb (section 4.2.2) it was shown that with a constant force there is a resulting velocity for a null input velocity. To understand how gravity can affect the overall speed of the robot a null velocity was sent to the legs

and then the pitch angle of the surface was changed as before. Results are shown in Figure 4.19(b). With legs still swinging when required and with sufficient inclination, the robot can walk for a long distance[5]. However, when the angle was small (below 15 degrees), the robot started walking backwards.

Swing ends when the leg has started supporting the body, which is detected by increase in load. Because the position is not known, legs keep moving down until certain load is detected, as explained in section 4.1. The position of the tarsi during AEP is in front of the BC joint and therefore resulting forces are pointed backward. This force towards the back is done mainly by middle and front legs, whereas the hind leg's force is small and to the front. Consequently, with every swing-stance transition there is a short force pulse backwards, which in turn moves the robot. In Figure 4.19(b) results without legs swinging are shown with ($*$), it is clear that now the body always moves forward. The geometry of the body still influences the direction of walking, therefore, if subordination is not null there is always some movement. Nonetheless, it is reasonable to assume that if the insect does not intend to move -zero speed- it would simultaneously set global subordination to zero.

4.5 Summary and Discussion

Legs receive limited information from other legs and they are not centrally controlled. Most of the inter-leg shared information relates to leg coordination, therefore, legs on stance require a positive feedback mechanism to synchronise their movement. However, hypotheses presented in chapter 3 about the negative feedback being always present and legs opposing external and internal forces have been further supported herein. The latter assumption is based on experiments shown in chapter 3, which demonstrated that legs always try to follow their own direction regardless of other legs' intention.

The controller proposed herein has a negative feedback velocity based control that shifts its set-point according to the deviation on individual joints. As a result, the system can behave like a position or velocity controller and simultaneously controls its own state while following external and internal forces according to its own subordination. Furthermore, individual joints can be activated more than others, therefore, joints can deviate the direction of the leg by means of a simple mechanism. The robot simulation demonstrated that this controller produces reliable walking patterns. External

[5]Distance not shown in figure.

forces, such as gravity, influence walking without compromising the initial intention.

4.5.1 Controller Equation

The swing trajectory does not affect walking in this model other than by protracting each leg to a position from which it can make a stance movement of sufficient duration. Hence, a simple control strategy for the swing was used, in which the α and β joints are moved to predefined target angles when the leg is unloaded. The leg is lifted and lowered by the β joint; and the end of the swing is signalled by the leg load when the tarsus contacts the ground. The AEP is not explicitly calculated but results from the intersection of the joint angles and the ground surface. To improve the leg coordination during turns, caused by altered direction during stance, predefined angles at the alpha and the gamma joints retract and extend, respectively, proportional to the angle ϕ. This results mainly in AEP falling closer to or away from the body in the outer and inner leg respectively.

The swing controller has similarities with swingnet3 as described in (Schumm and Cruse, 2006) because the lifting and lowering of the leg is solely controlled by the CT joint. The swing controller is clearly not based on biological data. However, it allows influencing the position of the AEP by specifying the BC and FT joint anterior target positions, which could easily be set by the anterior leg. The AEP is the only variable that affects walking because there are no obstacles in the arena, and thus aerial trajectory does not affect results presented herein. It would be vital for a complete model to include a more sophisticated swing controller to deal with obstructions. The time legs take to move forward is more closely related to a robotic implementation, which supports results for a real construction.

It was also demonstrated that the stance controller allows individual legs to coordinate with each other. Simultaneously, it is still possible to control their direction and always respond as a negative controller. These were the three requirements proposed for a leg controller at the beginning of section 4.2. This stance controller is supported by concurrent work done by Schneider (2006). First, equation 4.5 needs to be converted to motor velocity instead of position and scale the variables to the sampling frequency of the controller. This is shown below,

$$\dot{\theta}_{M,t+1} = \dot{\theta}_{sp,t} + s\,\theta_{d,t}/\Delta t + \Phi\kappa/\Delta t \tag{4.7}$$

$$\dot{\theta}_{M,t+1} = \dot{\theta}_{sp,t} + s'\,\theta_{d,t} + \Phi\kappa' \tag{4.8}$$

Equation 4.8 is the velocity sent to the motor, where $s' = s/\Delta t$. Schneider et al. (2006) proposed a number of local positive velocity feedback (LPVF) controllers similar in operation. Early versions of LPVF did not allow power generation by joints, and thus were not as relevant for the control of walking. However, the following equation was used on a single leg preparation (Schneider et al., 2005c),

$$\dot{\theta}_{M,t+1} = K\dot{\theta}_{j,t} + \rho\,\theta_{b,t} \tag{4.9}$$

where global speed is controlled by moving K, $\theta_{b,t}$ is the difference between motor and segment position and $\rho = \{0,1\}$ (named *relaxation*) was made zero when joints do positive power[6]. Concurrent work on the crank turning experiment led Schneider (2006) to propose a power controlled continuous relaxation controller,

$$\dot{\theta}_{M,t+1} = K\dot{\theta}_{j,t} + \Big(1 - \frac{P_j}{Pmax_j}\Big)c\,\theta_{b,t} \tag{4.10}$$

$$\dot{\theta}_{M,t+1} \approx K\dot{\theta}_{j,t} + c\,\theta_{b,t} \tag{4.11}$$

where P_j is the power of the joint. Note that the *relaxation* factor c and the subordination value s are therefore equivalent because both control the influence of external forces affecting joint velocity. However, the main difference between equation 4.8 and equation 4.11 is the velocity set-point used to update the motor position. On equation 4.11 information of other joints is not included and the control of speed is based on its current speed and variable K (positive feedback).

It is advantageous to control $\dot{\theta}_{sp,t}$ instead of $K\dot{\theta}_j$, because it allows exact manipulation of the whole leg. As a side effect, this approach requires shared information among joints within each leg and geometrical information. For instance, segment length, form and position. However, as mentioned in section 2.4, it is neurophysiologically plausible for joints within each leg to share information about position and velocity. The advantage of positive feedback is to allow all joints to synchronise with each other without requiring a central controller. Additionally, it also allows legs to overcome uncertainties on the terrain and other minor obstacles. These advantages however, can be accomplished by the subordination parameter without needing to feedback the velocity, i.e., without sustaining ongoing motion. The disadvantage of this approach is that the subordination needs to be controlled depending on the behaviour at hand. Whereas models proposed by Schneider et. al. are self-adapting and require only local joint

[6]This could take as much as half of the stance phase.

position. The latter model has been tested on a single leg preparation and is yet to be implemented in a hexapod.

The sensor arrangement was slightly different to that presented herein, because initial models depended on sensors measuring the bending of the joint. Latest versions (Schneider, 2006; Schneider et al., 2007) resemble more the biological system by measuring the joint position instead, in the same manner as in Figure 4.3. Nevertheless, both single leg controllers have a negative feedback mechanism that is controlled by servomotors in the robot or by muscle elasticity properties in the insect. However, the reference for the model described in (Schneider et al., 2005a, 2006) is always based on the current joint velocity regardless of the bending. Therefore, the joint will try to restore its velocity from external influences only during the onsetting of the disturbance and then it will support that new motion. For instance, in the experiment presented in section section 4.2.2, after the pulse force, the tip of the limb would continue to move in the direction it was moving shortly before the force was removed. On the other hand, the controller proposed herein resumed its original intention after the force was removed and it always tried to sustain that motion. Therefore, the reference in the controller described by Schneider et. al. is so influenced by external influences that the negative feedback part of the controller is only used for transient disturbances and not to sustain voluntary motion. This is an important requirement because, as suggested in section 3.8, legs need to sustain its own direction independently of other legs and external forces. The important element of the controller proposed herein is the negative feedback mechanism in combination with an active and continuous velocity reference.

The balance between positive feedback (energy saving) and negative feedback (energy investment) depends upon the task. Therefore, it is more suitable to have negative feedback as a central controller for those tasks that repeatedly change direction or require increase in load. Furthermore, initiation of motion is better coped with by those with negative feedback. It was shown that positive feedback has advantages, however, it was proved that negative feedback is a necessary mechanism for a dynamic system. Otherwise, positive feedback based controllers would require large inputs of energy to continuously get the desired behaviour.

4.5.2 Matching Insect Behaviour

The implemented controller cannot be easily categorised as a position or velocity controller because it depends on the variable parameters specified for each joint. For

instance, if the internal angular velocity is set to zero $\dot{\theta}_{sp} = 0$ and subordination is low $s \ll 1$ it will respond like a position controller. Furthermore, if subordination is non-zero $s > 0$, even if $\dot{\theta}_{sp} = 0$ the leg will be eventually moved to a new resting position and it will not return to its original position. In the first situation, the leg will be more precisely described as position control based; this reflex reversal agrees with results on the standing insect (Cruse et al., 1992). On the latter situation, it responds like a velocity controller; this also agrees with results on the stick insect (Bartling and Schmitz, 2000). Furthermore, higher subordination values explain the reversal reflex investigated for the stick insect (Bässler, 1976). Therefore, by adjusting the subordination parameter it is possible for the proposed controller to respond like the insect.

Most of the characteristics described above require changes on the subordination parameters. However, the scope of this thesis could not include that situation. Variability of the leg's subordination should not be controlled at a lower-level because resulting deviations are not perceptible at the leg level. This does not imply that there should be a dedicated central controller, it only suggests that subordination depends on the initial intention. If the insect intends to camouflage, that decision is more likely to come from the brain, thus a low subordination (position control) should be included on that command. This command could depend on the intended speed, but that should also originate in the brain.

4.5.3 Plausibility of Real Robot Implementation

Controllers presented herein should be implementable with available hardware for the real robot. It is common practice not to construct robots from scratch, but rather to integrate specialised modules. For instance, it is common to use a dedicated servomotor instead of joining an encoder, a motor and a gearbox. Furthermore, servomotors that control position and not velocity are chosen when motors are not intended to rotate more than 2π, because these are easier to control. As mentioned in section 4.3, this was also more stable to implement in ODE. It is important to note that it is still possible to control velocity with position controlled servomotors, and vice versa.

Above it was shown that the absolute value of the deviation is not required, as the subordination parameter scales it. The deviation can be estimated either with the difference between motor position and segment position, or by the difference between target position and motor position. However, the former is easier to calculate provided a proper sensory implementation and frees the motor from constant excess in load. If

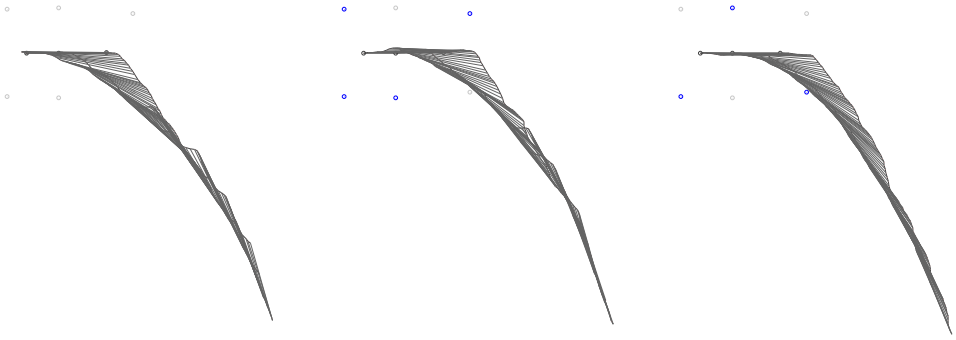
the segment does not allow displacement relative to the motor, the displacement can still be detected with a stress gauge or alternative specialized sensors. On the other hand, implementation can be done entirely in software, provided one has access to the internal servomotor controller.

The disadvantage of the controller presented herein is that a mechanism for setting the subordination value has not been proposed. Some range of values have been suggested for various scenarios, for instance, the standing animal is better described by zero subordination at joints. However, (Schneider et al., 2007; Schneider, 2006) have proposed a self-adjusting mechanism based on (Cruse et al., 2004); this mechanism is in particular for the standing animal.

Theoretically, both implementations are equally feasible. Nonetheless, real implementations often add complications. For instance, hardware is not easily tuned to behave differently, such as changing the spring constant. Additionally, adding springs to each joint increases the number of sensors required. On the other hand, the spring mechanism shown in section 4.3 represents already a robust and reliable position controller. The maximum torque it produces is not constrained as with the motor, it is more stable, energy free and its response time is difficult to match.

The main difference could be the time it takes the software implementation and the mechanical arrangement to control to the desired position. The spring is much faster and usually for springs with high spring constants oscillations are reduced without instabilities. Thus, the deviation detected using the spring box is more likely to be dependent on the external force and not intrinsic passive response.

Unfortunately, this research did not have the time to test the controller in a physical robot. As previously mentioned, all sensors and actuators were considered based on a real robot. The motors at the joint behave like servomotors and operate within their operational frequency. Similarly, sensors sample at reasonable frequencies and are not taken directly from the simulation. Nonetheless, one of the best arguments supporting the controller implementation in a robot, are the dimensions and weights used for the robot simulation. To support this claim further, the simulation was tested for larger variations in weight and size. Qualitative results are shown in Figure 4.20 for robots of different sizes and in Figure 4.21 for robots of different weights. These results suggest that the controller could work when large forces are considered. All inputs and outputs were scaled linearly with respect to the size and weight of the robot. However, the only two parameters that had to be calibrated manually were those specifying the damper and spring constants at the springs of joints. Once done, the subordination needed not

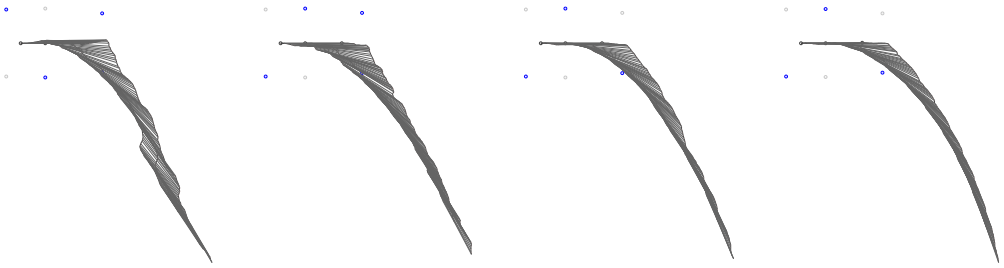


(a) size=1.82/4 m (b) size=1.82/2 m (c) size=1.82 m

Figure 4.20: Robot simulation at various sizes

From left to right sizes for the robot were 1.82/4=0.455 m, 1.82/2=0.91 m and 1.82 m. Trajectories were scaled accordingly to match the robot on the right. The robot is visually targeting an object at 60 degrees.

be changed because it depends on the angular deviation. It is important to note that the smaller and the lighter robot versions were relatively more unstable. Furthermore, the heaviest robot trajectory does not seem to rotate sufficiently fast. Nonetheless, two important features have been demonstrated. First, the controller is easily calibrated for various sizes and weights. Secondly, the controller works in dynamic systems and thus it is very likely to work on a real robot.



(a) weight=66.56/8 kg (b) weight=66.56/4 kg (c) weight=66.56 kg (d) weight=66.56x2 kg

Figure 4.21: Robot simulation at various weights

From left to right weights for the robot were 66.56/8=8.32 kg, 66.56/4=16.64 kg, 66.56 kg and 66.56 x 2 = 133.12 kg. The robot is visually targeting an object at 60 degrees.

Chapter 5

Thoracic Differentiation for the Control of Turning

In this chapter, the behaviour of different versions of the controller presented in the previous chapter is compared for the behavioural task of visual targeting.

First, the behaviour of the controller is analysed when only front legs are used for targeting with very little front leg subordination and middle and hind legs highly subordinated. This means that the front legs can influence the other two thoracic segments to drive turning without any explicit control of turning in these segments. The body trajectories produced are similar to those seen for the intact insect. However, this model cannot reproduce the insect behaviour shown in section 3.6, when the front tarsi are blocked. The robot simulation was then modified to behave like the insect with front tarsi blocked.

Within this chapter, it is also assessed whether it is possible to get the behaviour described in section 3.6 by using identical controllers for hind and middle legs. One scheme, proposed by (Kindermann, 2001), uses only feed-forward control with turning produced by biasing the alpha joints. A similar controller is analysed that combines velocity and feed-forward control, but with turning accomplished by altering the speed on each side, or biasing the alpha joints as before. As will be discussed, none of these produces satisfactory trajectories or leg contributions comparable to the experimental data.

In section 5.4 the additional turning control parameters k^{γ} and k^{α} introduced in section 4.2.1.1 are used for testing the turning behaviour in the condition with front tarsi blocked. Simulation results indicate that it is necessary to have the proper balance between the metathorax rotation and the mesothorax lateral movement for this model

to succeed. This calibration involves two types of parameters, active and passive; these are analysed in turn in sections 5.4.1 and 5.4.2 respectively.

Finally, in section 5.5, the full model under normal conditions is tested (i.e. without the front tarsi blocked) and it is shown that this produces a better match to the animal behaviour.

N.B. Some results in this chapter have been previously published in (Rosano and Webb, 2006, 2007).

5.1 Leg Coordination for Turning

The leg coordination is not centrally calculated. It emerges from communication between neighbouring legs. Information transmitted and received is very limited if it was intended to analytically solve the problem of coordination. However, by following a set of rules proposed by Cruse (1990) it is possible to reproduce the behaviour found in insects (Dürr et al., 2003). Also, different gaits emerge without being explicitly set, as a consequence of increased or decreased walking speed. Each leg has a step selector that determines, based on neighbouring signals, if this leg is in stance or in swing phase. These rules are graphically represented in Figure 5.1.

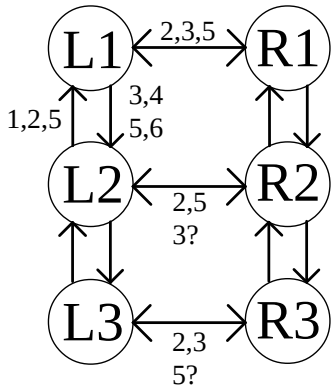


Figure 5.1: Coordination rules

Arrow indicated the direction of the influence. Refer to text for explanation of rules.

The coordination of legs was necessary to test the hypotheses on the control of stance and single leg control. The main priority was to keep the control of coordination

as similar to known behaviours of the insects as possible. However, it was also very important to reduce locomotion problems that would depend only on leg coordination. Therefore, some mechanisms were reinforced and are probably redundant for practical applications. However, it was intended not to have complications other than those caused by the single leg controller. Additionally, some variables need to be adjusted to work with variable stance direction. This necessity has been previously noticed for omnidirectional legged walking (Fielding and Dunlop, 2004).

Before reinforcing these rules it is important to understand what these rules do for coordination. One can divide them into two types of rules, those introduced to prevent problems with individual legs, and rules particularly preventing body collapse. The latter is focused on stance-swing transition and can be further divided into those supporting swing (excite) and those obstructing it (inhibit).

5.1.1 Rule's Purpose

The rules proposed by Cruse (1990) are:

1. Swing phase inhibits start of swing on anterior leg
2. Start of stance excites start of stance on anterior leg
3. Caudal position excites start of stance on posterior leg
4. Position of anterior leg influences position at end of stance ("targeting")
5. (a) Increase in resistance increases force
   (b) Increase in load prolongs stance
6. Treading-on-tarsus reflex

Legs are mechanically constrained to work within a certain range, beyond which they cannot sustain motion or the body. In straight walking this depends on how caudal the tarsus is with respect to the body-coxa joint. By inducing a new stance phase in a caudal leg with rule 2 it is indirectly stimulating its own stance-swing transition by a consequence of rule 3. It could be assumed that the nature of rule 2 and rule 3 is to prevent legs from being stuck, which eventually would impede walking.

On the other hand, coordination rules need also to consider body stability for avoiding falling down. The simplest analysis scenario is that of six fixed legs that can be only

up or down and whose body centre of mass is also located at the centre of the body. If the body is considered a rectangle with three fixed equidistant legs on either side, then it is possible to restrict the total number of leg positions to sixty-four. Assuming this simple system, forty-six out of sixty-four positions are not statically stable for this simple model (71.8%). However, if no ipsilateral neighbouring legs are allowed up at the same time, 39 unstable positions are avoided. Therefore, 57 positions (89%) are correctly evaluated as stable or unstable by rule 1 alone.

The variables on which all coordination rules are based depend on the platform these are controlling. For instance, leg distance threshold is calibrated upon the size, and the load of rule 5 depends on the robot weight. Furthermore, some are sufficient for straight walking but performance decreases when legs move in different directions, e.g., rule 3 dependency on caudal position. Therefore, some of the rules were augmented to increase coordination robustness and to simplify their implementation. This would allow focusing on the control of stance for normal walking and turning as well. The principles upon which these rules operate were the same, and should match for the simplest tasks, e.g. straight walking. For instance, rule 3 used additional sensory states to operate, however for straight walking it reduces to be proportional to the caudal tarsus position.

5.1.2 Rules Augmented

Coordination rules have already demonstrated their efficiency for the control of walking. However, these require a dedicated calibration process and it is not convenient to repeat the calibration process for each particular task. Furthermore, it has been proposed (Dürr, 2005) that calibration is not fixed and that the strength of each rule is modulated according to the walking task. However, due to time constraints, it was decided instead to compensate by augmenting rules in the hope that variable coupling strengths would not have to be included. In addition, it shall be demonstrated that while turning leg coordination changes without explicitly adjusting rules' coupling strengths.

5.1.2.1 Rule 1, Swing phase inhibits start of swing on anterior leg

It was mentioned before that, for the simplest model, rule 1 correctly evaluates stability on 57 of the 64 possible leg combinations. However, if rule 1 is further applied to not allow contralateral neighbouring legs up at the same time, it would classify 63 position combinations correctly; only one of them would be incorrectly labelled as unstable.

Therefore, by including that additional condition the chances of getting an unstable leg configuration are almost eliminated. Two contralateral legs swinging is not normally seen in the insect. Front legs sometimes do it when changing direction abruptly, however, the centre of mass is located behind the metathorax so this is still stable. Also, when crossing large gaps the body is only supported by front and hind legs (Bläsing and Cruse, 2004b). Forward speed in this situation is almost zero, and coordination rules in general are modulated differently. It is possible for this additional condition to emerge as a secondary effect by rules that do apply contralaterally, however, this redundancy was found to be beneficial to the coordination problem.

5.1.2.2 Rule 3, Caudal position excites start of stance on posterior leg

Front leg tarsi move almost sideways for most of the initial phase when the insect is doing a tight turn. Rule 3 would not evaluate properly the desired condition because it depends on the caudal position of the leg, i.e. distance parallel to the body. Furthermore, the average speed is also different, particularly for hind legs, hence the time on stance cannot replace caudal positioning or distance travelled. It is possible to calculate the distance travelled instead of the caudal position. This is a good estimator, but it has to be complemented because the distance a leg can travel depends on the direction it would follow.

The ability for a leg to support and move the body decreases as the distance it has travelled also increases. Additionally, joints can be directly used to signal when legs are reaching the end of their possible movement. Each joint has a maximum and minimum value. By indicating their proximity to this boundary it is possible to predict future problems. This was implemented in all joints.

A particular problem with front legs is that for large changes in direction[1] in motion towards the body, the tarsi could eventually cause the knee to touch the ground. To avoid this, the angle of the tibia segment was also included in rule 3. The angle between the ground and the tibia indicates where the FT joint (knee) is positioned. This angle can be estimated as $\theta^{\gamma-ground} = \pi/2 + \psi' + \beta + \gamma$. Therefore, a leg's contribution was diminished as this angle approached zero. This additional parameter was in fact rarely used for the trajectories shown in this section. Nonetheless, hereafter it proved to be necessary when the simulation turns at 180 degree.

A similar approach based on increasing rule 3 to detect mechanical constraints in

[1] More than 90 degrees

legs was proposed by Fielding and Dunlop (2004). The term ‘restrictedness’ of legs was introduced to calculate how likely legs were of becoming stuck. This was based on six mechanical and logical constraints. Three of the six parameters are directly related to individual joints being close to their maximum and minimum values. However, the others are less likely to be biologically plausible, e.g., tarsus-coxa plane projected distance.

5.1.2.3 Rule 5b, Increase in load prolongs stance

This indicates that the centre of mass is close to the body-coxa joint of this leg and therefore it is important not to remove support (Cruse, 1985b). Moreover, in the insect lateral forces can be independent of normal forces because the tarsus holds the ground. Nonetheless, the robot tangential forces depend exclusively on frictional forces to walk, i.e., $F_T = \mu F_N$. Consequently, it is essential for a leg without a gripping mechanism to have load, not only to support the body, but also to move the body. Therefore, rule 5b was augmented by inducing stance-swing transition when leg load was low. This has also been suggested as a responsible parameter that triggers swing in the stick insect (Cruse, 1985b; Schmitz, 1993).

5.1.3 Calibration

Variables controlling the coordination of legs are not directly taken from sensors, i.e. most require an intermediate stage. Furthermore, calibration was not easy because of the amount of variables involved. Also, the morphology of the system would produce different calibration values for each of the thoracic segments. Leg transition depended upon various parameters and leg states that were not of similar units, e.g., step transition depended on load, distance, angles, etc. All these variables needed to be combined to produce a binary state that would indicate the leg to remain on stance or to swing.

In order to treat all variables equally, a fuzzy logic based filter was implemented. The output is normalised based on the variable average range of values, its minimum and its maximum. These were found by moving the robot simulation in different directions; variables were updated online. Essentially, this procedure allows the system to estimate what “low” load or “large” distance was. The value given was a continuous value between 0 and 1, which corresponded to “low” and “high” respectively.

All variables used for calculating step transitions were normalised and then input to a probabilistic model that calculated the likelihood of a leg changing from stance to

swing. The probability of the leg *Leg* to swing based on the neighbouring leg Leg^*, was calculated with the following equation,

$$\begin{aligned} P &= Excite - Inhibit && (5.1)\\ Inhibit &= 1-(1-Leg^*_{swinging}\,w_0)(1-load_{high}\,w_1) && (5.2)\\ Excite &= 1-(1-Leg^*_{distance}\,w_2)(1-load_{low}\,w_3)(1-leg_{forced}\,w_4) && (5.3)\\ leg_{forced} &= \alpha_{high}+\beta_{high}+\gamma_{high}+\theta^{\gamma-ground}_{low} && (5.4) \end{aligned}$$

Negative values of the variable P were treated as zero. All variables were normalized, including distances, e.g., $Leg^*_{distance}$ is the normalised distance of the neighbouring leg. The weights w_n control the influence of each value given by the fuzzy logic filter. This probability was checked at every time step, thus the probability was adjusted so as to not depend on the simulation frequency. The probability p of an event happening increases the more frequently the condition is checked. If the condition is tested at a frequency f, and the probability of the event to happen needs to be P within s seconds, then p must be set to,

$$p = 1-(1-P)^{s\,f} \qquad (5.5)$$

The probability of an event happening is modelled by a Poisson distribution, which similarly resembles the probability of a spiking neuron. Although parameters for deciding leg phase transition were easier to set for forward walking, the difficulty increased when including those for turning. Consequently, manual calibration became less predictable. In addition, some of the rule additions mentioned above were not intuitive. Therefore, a genetic algorithm was run to explore the parameter space for straight walking as well as for turning.

5.1.3.1 Genetic Algorithm

The genetic algorithm (GA) was programmed in Java™. It consisted of two main programs: the *Experiment* and the *Server*. The total number of experiments to run was divided onto different computers and was administrated by a central program. Each computer was responsible for running a population of insects with certain parameters and collecting results. The central program was responsible for the genetic algorithm itself, i.e., combine genes, create generations, etc. The *Experiment* program was totally

independent of the GA, therefore different generations could be run in parallel on many computers. The *Server* was also responsible for controlling the different processes.

The next generation was calculated only when all information was collected back from the previous one. Therefore, the average speed was that of the slowest computer divided by the number of computers used. Experiments were divided among computers indiscriminately because each experiment was evaluated individually.

Population size was set to 20000 individuals for most sets of generation runs, which was usually set at 100 generations. At every generation 1% of the fittest parents were cloned to the next generation. The remaining parents genes were crossed over (95%) and mutated (23%).

Gene encodings were not binary but use continuous values [0, 1] in order to reduce chromosome size. Each of these values corresponded to a gene, thus chromosome size was that of the number of variables to find. During crossover individual variables did not change, i.e., entire values were swapped. For this reason mutation rate is larger than usual to allow for thorough sampling of the parameter space. Consequently the building block hypothesis (Mitchell, 1998) does not hold and this would be reflected in the overall performance.

Walking speed was constant for all experiments but at a random continuous direction. A fixed axis of rotation was set to the left or right of the hind inner leg, velocities and leg directions were centrally controlled. Turning at this point was largely unknown, results were based on (Rosano, 2004). The fitness function was the added contribution of each leg, updated every time step. This function was proportionally increased with the velocity and load during stance. The fitness function was decreased when legs could no longer move and proportional to sensed load during swing. This is shown in the following equation,

$$Fitness = \sum_{t=0}^{T} \sum_{i=1}^{6} r_{\vec{L}}(t)_i + Load_{stance}(t)_i - Load_{swing}(t)_i - Leg_{stuck}(t)_i \qquad (5.6)$$

where t is the time and T is the total time. The fitness was the sum of all legs, as shown by the subindex i. Experiments terminated below certain negative fitness to increase computational speed, and for experiments in which the insect body touched the ground.

5.1.4 Leg Coordination Results

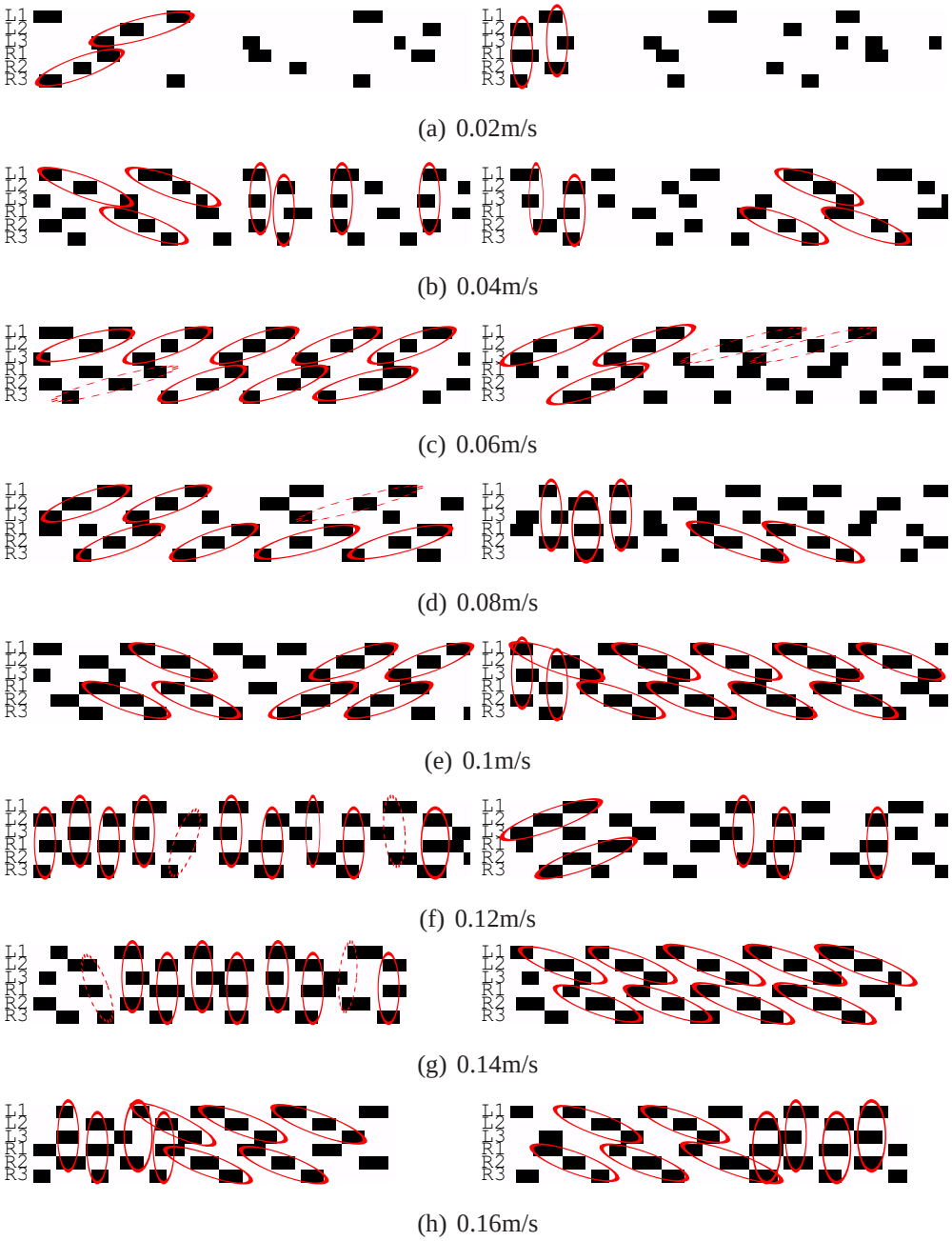
Parameters found with the GA resulted in a very stable system, however common gaits were not as stable as with the insect. For each speed the simulation was run three times. Two of these stepping results for each speed are shown in Figure 5.2. Particular emergent patterns are marked for easier visualisation. Note how different gaits emerge within the same experiment and other experiments barely have any gait. The average number of legs on the ground showed a strong dependency on the insect speed and tetrapodgait was the most common. Figure 5.3 shows results of straight walking at leg speeds from 0.02m/s to 0.16m/s. On the top left corner, the lag between front leg and ipsilateral hind leg versus the period of the former is shown.

A unitary slope indicates a tripod gait whereas a non-unitary or non-linear dependency on the phase indicates a tetrapod gait. On the top right corner it is possible to see that lag-phase points accumulate in two areas; shown by the two peaks. Therefore, the robot system seems to show both gaits, nonetheless emergence of these gaits does not depend on the speed as it should. This is shown in the bottom left plot. At all speeds the systems seems to use both gaits indiscriminately. Nonetheless, the tripod gait is more likely to be present at higher speeds. This is shown on the bottom right plot, where the average number of legs on stance is shown as the speed varies. It might be possible that at higher speeds for the robot the tripod would be more likely than the tetrapod. However, the speed of the robot simulation reaches a limit at 0.16m/s and the motor cannot go faster.

5.1.5 Summary

Coordination of legs is not centrally controlled, it emerges from rules proposed by Cruse (1990) that secure global stability by assuring local stability. These rules needed to be adjusted to the model proposed herein because some of the variables were not compatible with turning. Furthermore, it was necessary to increase reliability to reduce problems dependent on leg coordination when testing the single leg controller. For instance, Rule 1 was augmented by inhibiting contralateral legs as well; Rule 3 was partially based on distance travelled and not on caudal position; Rule 5b induced stance-swing when load of the leg was low. Coordination rules are probabilistic and variables are pre-processed by a fuzzy logic-like filter.

The calibration process was performed using a genetic algorithm run on multiple computers in a client-server network architecture. Leg coordination was not as stable



(a) 0.02m/s

(b) 0.04m/s

(c) 0.06m/s

(d) 0.08m/s

(e) 0.1m/s

(f) 0.12m/s

(g) 0.14m/s

(h) 0.16m/s

Figure 5.2: Stepping patterns at different walking speeds

Swing is represented by black areas and stance in white. The first three rows from the top are legs on the left side, L. The last three are legs on the right. In each side, legs are positioned at the prothorax (1), mesothorax (2) and then metathorax (3). Dotted lines indicate weak temporal dependency for that gait (cf., Figure 2.6). Two representative sequences are shown for each walking speed.

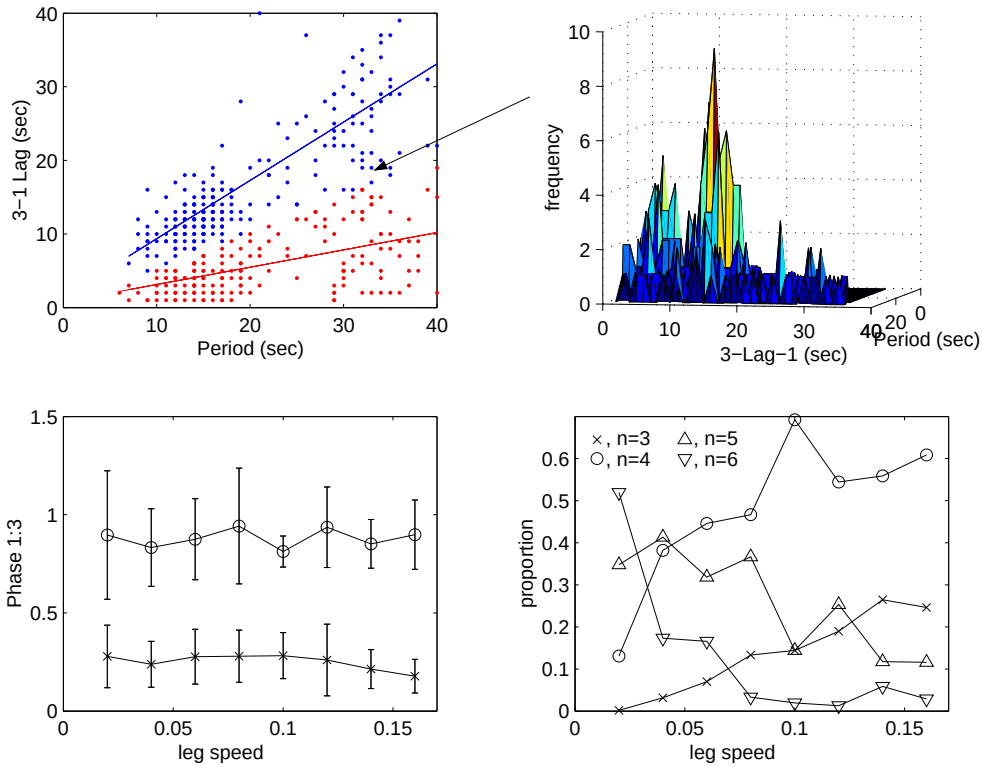


Figure 5.3: Statistics of walking gaits

The lag between ipsilateral front and hind leg, shown on the top left, resulted in two different phases, shown by the two peaks on the top right plot. The arrow shows the direction of view set on the top right plot on data of the top left plot (azimuth 100.5, elevation 4). Phases between legs 1 and 3 (front and hind) were constant, as shown on the bottom left plot. The proportion of legs in stance as the speed changes is shown on the bottom right.

as with the insect, however, it showed two of the most common gaits and an increased probability of favouring tripod gait as the speed increased.

5.2 Results with passive middle and hind leg control

The following experiments used the model described in section 4.2 but without the additional active turning contribution of the middle and hind legs (as described in equation 4.4). For the front legs the subordination parameter was zero, $\vec{s}_{pro} = 0$, i.e. they follow a straight line toward the target under direction control alone, without any influence from external forces. The middle and hind legs are uninfluenced by the target but with a high subordination, i.e., they would continue walking forward but are principally influenced by front legs. It was found (through empirical testing) necessary to vary the subordination parameter for different turn sizes: for target angles below 60 degrees ($\vec{s}_{meta} = [0.10, 0.01, 0.15]$ and $\vec{s}_{meso} = [0.40, 0.01, 0.20]$) and for larger angles ($\vec{s}_{meta} = [0.15, 0.01, 0.35]$ and $\vec{s}_{meso} = [0.50, 0.01, 0.50]$).

The robot model was made to turn at angles from 20 to 90 degrees by increments of 10 and due to the symmetry of the system all turns were made to the same side. Because gait coordination is probabilistic, and every time a different pattern was found, three runs were taken for each angle, for a total of 24 turns. Runs were stopped once the body angle was within 5 degrees of the target and the metathorax was aligned with the prothorax in the same direction.

Results from the simulation were analysed using the same approach as for the insect and are shown in Figure 5.4[2]. It can be seen that on average the prothorax, as for the insect, tries to achieve the target orientation very early during the turn and maintains it until the metathorax and body are facing in the same direction. However the prothorax does not have enough time to face in the same direction as the body before reaching the target. The behaviour of the metathorax direction and body angle are similar to the insect behaviour. However, the metathorax in the simulation varies more from the body direction. This means that the body is moving slightly sideways at the beginning of the turn. Results reported previously in (Rosano and Webb, 2006) are slightly different because only the body orientation was controlled. Introducing the target meant that the simulation had to correct more for deviations and had less time to do so.

Speeds of the prothorax and metathorax are shown on the right hand side in Figure 5.4. It can be seen that the speed of the metathorax is automatically decreased with

[2]Snapshots of this experiment can be found in appendix D.2.

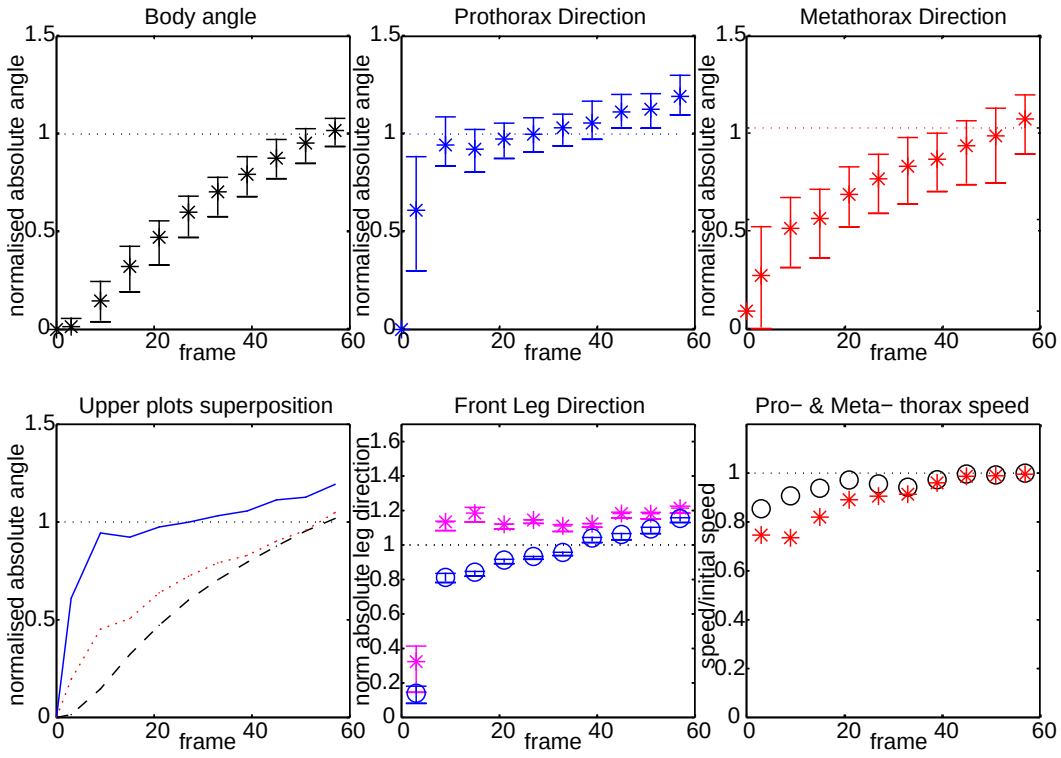


Figure 5.4: Body kinematics: Robot model, only the front legs actively turning

Absolute angles normalized to the target angle $\theta_T = 1$ (unitless) and the bars indicate standard deviation. Top left shows progress of the body angle θ_B, top middle is that of the prothorax θ_P and the top right is the metathorax direction θ_M. Superposition of top plots is shown in the bottom left; metathorax θ_M with a dotted line; and the body θ_B with a dashed line. The bottom right shows changes in speed for the prothorax ($\circ$) and metathorax ($*$) relative to their velocity before the turn.The bottom middle shows the direction front legs follow relative to the initial heading, i.e., $\theta_B + \theta_L$; both legs are shown, inner front leg ($\circ$) and outer ($*$). cf. Figure 3.9 for insect.

respect to the prothorax without being explicitly controlled. However, the maximum decrease for the metathorax was approximately 25%; whereas the insect showed a decrease as large as 40%. The simulation was found, like the insect, to recover its original speed before finishing the turn. Figure 5.4 also shows the direction front legs follow relative to the target. However, their behaviour is not as smooth as that shown by the insect in 3.9. This suggests that front legs should perhaps be partially influenced by external forces. However, increasing the subordination parameter for the front legs was found to impede targeting.

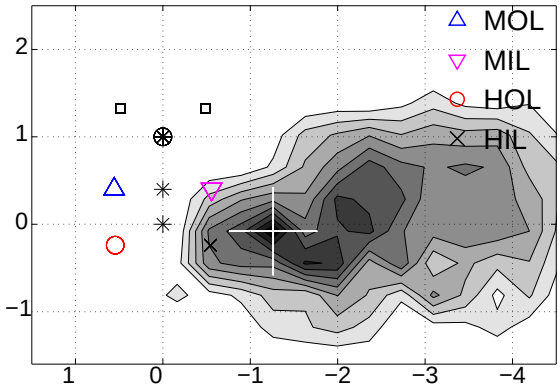


Figure 5.5: IAR: Robot model, only the front legs actively turning

Cumulative plot of the instant axes of rotation (IAR). The contours represent the distribution of the IARs, the axes is the distance normalised to the distance from prothorax to metathorax. The white cross shows the average axis of rotation (AAR). The three black asterisks (∗) represent the three thoracic segments and the prothorax (front) is encircled. All turns were properly transformed to the right before statistics were calculated. Middle 'M', hind 'H', inner 'I' and outer 'O' legs 'L'.

The accumulated instant axis of rotation (IAR) of the simulation is shown in Figure 5.5. The AAR for the simulation is, as for the insect, found between the hind and middle segment. However, it is clearly located further away from the side of the body. One possible explanation is that rotation is not so close to the metathorax and the speed of the segment does not decrease as much, as verified by Figure 5.4. However, the opposite could also be the case, i.e., because the metathorax does not decrease in speed, rotation is moved away to one side. In either case, it was not possible to find an appropriate calibration with the parameters available to reproduce trajectories produced by the insect.

5.2.1 Why does the body rotate?

Previously it was mentioned that if front legs pull the body in the direction of the target ignoring the body rotation the middle and hind legs would eventually induce rotation. This is not similar to a four-wheel vehicle with steering in the front wheels. Legs could have moved mainly laterally if these had been more subordinated or controlled only by positive feedback. This could have resulted in a body trajectory perpendicular to the heading and with little rotation (lateral walking).

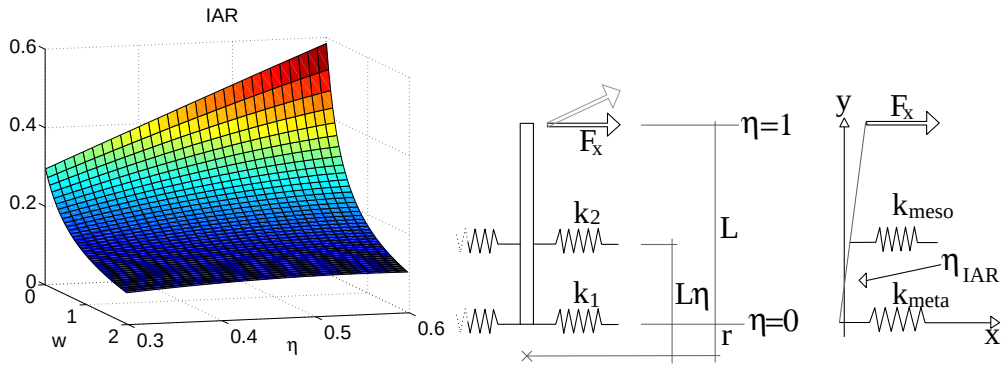


Figure 5.6: Why the body rotates when pulled by the front legs

The body rotates because opposing forces at the mesothorax and metathorax create a resulting torque.

The main reason for the resulting rotation is that the mesothorax and metathorax continuously oppose the direction of the prothorax and that creates a torque on the body. The axis of rotation depends on how much each thorax opposes the turn, however, it is more likely to be positioned close to the metathorax. Figure 5.6 shows a representation of the forces on the body. The prothorax is pulling the body toward one side and the springs at the metathorax ($\eta = \eta_{meta} = 0$) and mesothorax ($\eta = \eta_{meso}$), represent the opposing force. The relative distance of the IAR from the metathorax is given by the following equations[3],

$$\eta_{IAR} = \left[\frac{(\eta_{meso}-1)\eta_{meso}}{\eta_{meso}-1-w}\right] \quad (5.7)$$

$$k_{meta} = w k_{meso} \quad (5.8)$$

where k_{meso} is the equivalent spring constant at the mesothorax and k_{meta} at the metathorax.

[3]Details of this equation can be found in appendix C.1.

The position along the body of the IAR is given by η_{IAR}, e.g., $\eta_{meta} = 0$ and $\eta_{pro} = 1$ (Figure 5.6). This equation, plotted on the left of Figure 5.6, illustrates two important consequences. First, as w becomes zero, $\lim_{w\to 0}$, $\eta_{IAR} = \eta_{meso}$, the rotation gets closer to the mesothorax, i.e., strong opposition by this thorax attracts the IAR. Secondly, as w grows the IAR gets closer to the metathorax. $\lim_{w\to\infty}$, $\eta_{IAR} = 0 = \eta_{meta}$. The subordination values used for the previous experiment had $\vec{s}^{\gamma}_{meta} < \vec{s}^{\gamma}_{meso}$, thus, $k_{meta} > k_{meso}$, $w \gg 1$. By further decreasing $\vec{s}^{\gamma}_{meta}$ or increasing $\vec{s}^{\gamma}_{meso}$ the AOR can be moved closer to the metathorax. Nonetheless, low subordination values increase instability of the system because it follows all forces. High values would interfere with front leg's directionality. It is important to note that the axis of rotation can be controlled only when there are two flexible anchor points. Otherwise the AOR cannot be moved away from the anchor.

5.3 Eliminating some alternative models

An approach that has been proposed before is to control turning by introducing biases on every BC joint in the body and having all legs following every external force. For large curvatures, the trajectories produced are similar to the insect, but this method fails to reproduce tight turns. Furthermore, the contribution of individual legs, when front tarsi are blocked, is quite different to that found on the insect; these results are shown in Figure 5.7.

The trajectories followed by the prothorax, mesothorax and metathorax do not differ much (Figure 5.9) which is seen only for smooth turns by the insect (cf. Figure 3.14). A typical trajectory for this reduced model is shown in Figure 5.8(b) and the AAR for a set of experiments is shown in Figure 5.8(a). Because the AAR is not close to the body, and most importantly towards the rear, trajectories are large, smoother and usually failed to hit the target.

A version of WalkNet by Kindermann (2001) introduced a very small deviation in the AEP and biases in the BC joint. The turning trajectories this produces are similar to that shown in Figure 5.8(b). However, as mentioned in section 4.1, given the directionality the prothorax follows and its biological plausibility, the PEP is unlikely to be a control variable. Without the PEP as reference, stance trajectory is decided by ongoing movement. Therefore, the position of the AEP only improves the distance the leg can be on stance. It might be possible to decrease the curvature of the turn by modifying the AEP. However, as mentioned earlier, directionality of the prothorax requires con-

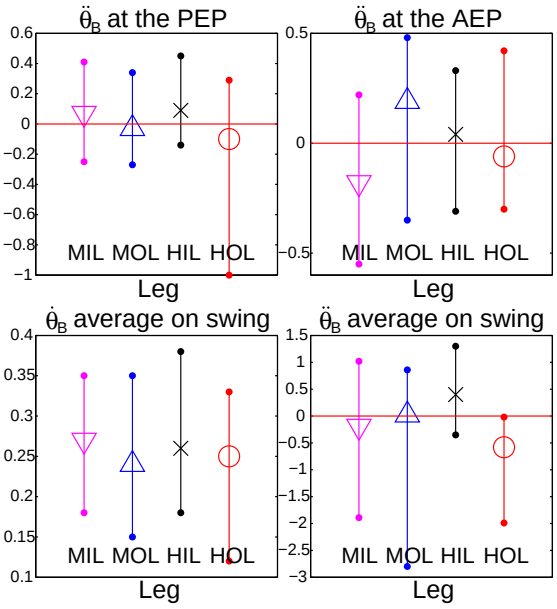


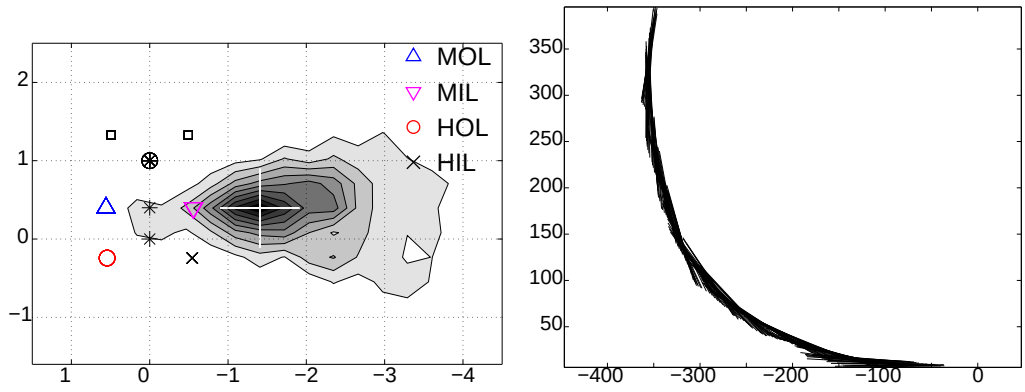
Figure 5.7: Swing statistics: Simulation with alternative turning strategies

Trajectories are not calculated with inverse Jacobian, turn is induced by introducing biases to the BC joint. The contribution of each leg to turning is the opposite to that found in the insect (Figure 3.14).

tinuous adjustment based on visual processing. For some sets of parameters, the leg contribution could look analogous to insect data, but matching trajectories and AAR is also critical, and was not possible using these methods of control.

A second alternative is to have only the mesothorax or only the metathorax actively controlled, with the other segment following passively. The active segment uses a single leg controller as described in section 5.2, i.e. using equation 4.3, and difference in speed on each side is implemented by directly changing the speeds ($r_{\vec{L}}$ on equation 4.2), or by introducing biases on alpha joints as before. This method is unlikely to work if the mesothorax is the active thorax because that would induce an axis of rotation close to the mesothorax. A representative trajectory is shown in Figure 5.10(a), note that the rotation around the mesothorax causes the metathorax to move in the opposite direction to the turn.

However, nor is the metathorax in an ideal position to produce all the necessary torque because it is at one extreme of the body. Therefore, the torque to produce rotation needs to be large, as with the prothorax in section 5.2. To produce the same body



(a) Accumulative IAR (b) Typical trajectory

Figure 5.8: IAR: Simulation with alternative turning strategies

Trajectories are not calculated with inverse Jacobian, turn is induced by introducing biases to the BC joint. On the left hand side the accumulative IAR shows that the average axis or rotation (AAR) do not get very close to the body. On the right hand side a typical trajectory is shown for targeting at 60 degrees.

acceleration, the metathorax needs a torque four times that needed by the mesothorax[4]. The simulation always produced smooth trajectories and the AAR was usually away from the body and at the level of the mesothorax. A representative trajectory is shown in Figure 5.10(b). If a high subordination is introduced in the mesothorax to aid the metathorax the system becomes unstable. This happens because the prothorax slips and the mesothorax follows all forces, the torque and inertia become too large for the metathorax. A representative trajectory is shown in Figure 5.10(c).

5.3.1 Discussion

Alternative mechanisms that would eventually result in leg trajectories shown in Figure 3.17 were tested. However, results suggest that both thoracic segments need to contribute to turning and that a more explicit trajectory was needed; particularly for the mesothorax. As mentioned in section 3.7.2, moving the γ joint will produce a deviation perpendicular to the body. The BC joint is always active and in theory, its direction $\hat{\alpha}$, and the FT direction $\hat{\gamma}$, form an orthogonal base. Therefore, the leg direction can be approximate to $\vec{L} \propto [\dot{\alpha}\hat{x}, \dot{\gamma}\hat{y}]$ and $\theta_{\vec{L}} = \arctan(\dot{\gamma}/\dot{\alpha})$ depends only on the relative angular

[4]Considering a cylindrical body and the mesothorax positioned at the centre and the metathorax at one extreme.

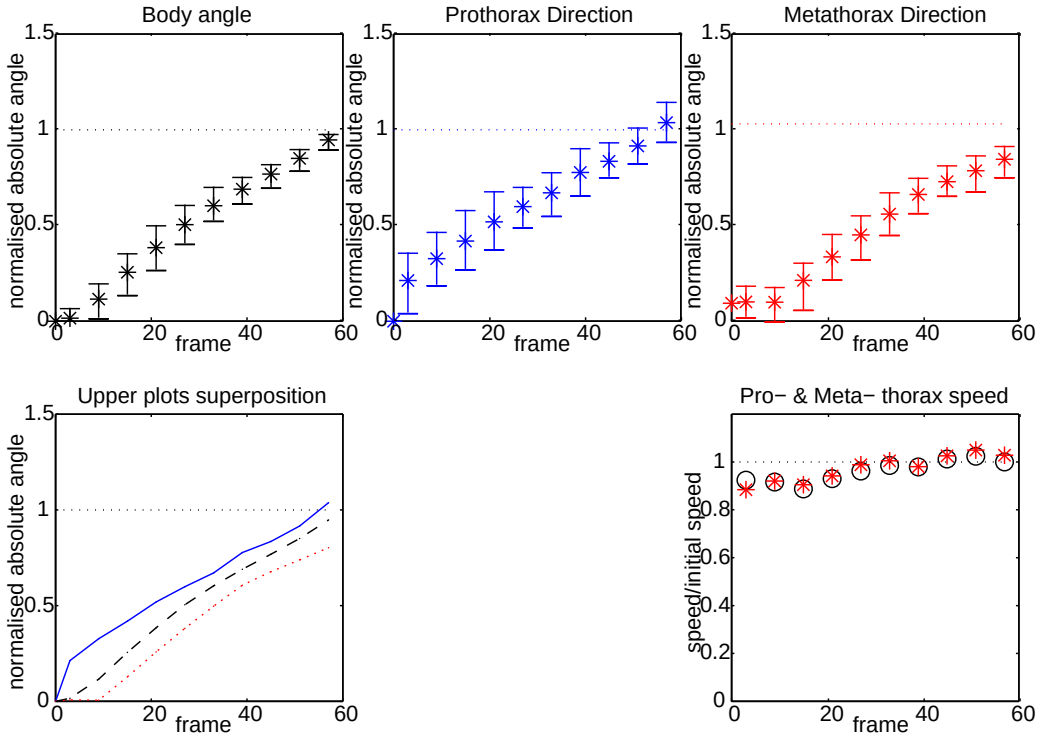
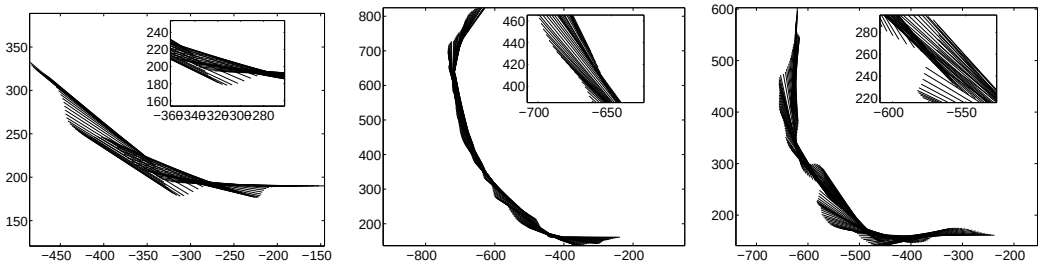


Figure 5.9: Body kinematics: Simulation with alternative turning strategies

Trajectories are not calculated with inverse Jacobian, turn is induced by introducing biases to the BC joint. It can be seen that each thoracic segment behaves similarly and with the same angle as that of the body; unlike Figure 3.12. Absolute angles normalized to the target angle $\theta_T = 1$ (unitless) and the bars indicate standard deviation. Top left shows progress of the body angle θ_B, top middle is that of the prothorax θ_P and the top right is the metathorax direction θ_M. Superposition of top plots is shown in the bottom left; metathorax θ_M with a dotted line; and the body θ_B with a dashed line. The bottom right shows changes in speed for the prothorax ($\circ$) and metathorax ($*$) relative to their velocity before the turn.



(a) rotation at the middle (b) long smooth curve (c) unstable turn

Figure 5.10: Representative turns for unbalanced rear thoraces

In (a) the mesothorax actively rotates and the metathorax is highly subordinated; AAR is closer to the mesothorax. In (b) the metathorax is active and the mesothorax is passive; the metathorax cannot cope with the necessary torque to rotate the body. In (c) subordination of mesothorax is increased further to aid the metathorax; turns become unstable.

speed. It seems possible to induce a lateral direction on the mesothorax legs by altering the BC joint and not the FT joint. The problem is that this works by decreasing the BC, this also decreases the speed of this segment, resulting in slower turning.

Consequently, it was demonstrated that models listed that do not differentiate the control of the different thoracic segments were unable to produce insect-like turning behaviour. Furthermore, those that differentiate but have only one thorax active cannot input the necessary energy in the system to rotate the body. Additionally, subordination values cannot be increased too much or the system becomes unstable.

5.4 Results with active middle and hind leg control

In section 3.7.2 it was suggested that there is active control of the direction of leg movements for the mesothorax and metathorax, i.e. the mesothorax moves sideways while the metathorax rotates. Note that this is intended as an additional mechanism to the front leg targeting discussed in the previous section, although for the experiments described here it is assumed the front tarsi are blocked, so forces from the front legs are minimal.

It was mentioned in section 5.2.1 that the resistance by the meso- and metathorax affect the position of the axis of rotation. That resistance is mainly controlled by the

FT joint in both segments. However, that model did not consider active roles of the middle and hind legs. One issue that arises then is how these active influences interact with the passive influences, i.e. how varying the subordination parameters s^{α} and s^{γ} will affect the turns produced. A second question is how these two influences should be balanced, i.e. what combination of parameters κ^{α} (modulating the metathoracic BC joint to produce rotation) and κ^{γ} (modulating the mesothoracic FT joint to produce lateral movement) will produce the results seen in the insect data. Each issue is examined in turn.

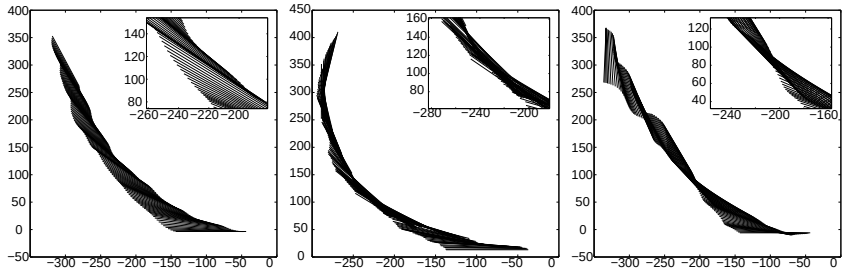
For this set of experiments, the environment and simulation settings were as described in section 5.2. However, the visual target for this experiment was positioned at 45, 60 and 80 degrees to the right, and the simulation was run 6 times for each of these angles; all 18 trials were repeated for each set of parameters and the results averaged. The front leg controller was not altered, but the friction on the tip of the leg was decreased to emulate the effects of blocking the tarsi in the insect. This means the front legs can still be moved and used for support, but they have little effect on the direction of movement, just as with the insect.

The main method for evaluating the parameter effects was to calculate the individual leg influences during turns, as shown in Figure 3.14. The aim was to produce qualitatively matching results, i.e. the appropriate relative influence of each leg, rather than match exact magnitudes, as these depend on detailed dynamics that differ between the insect and the simulated robot. Therefore, the first objective was matching all of the 16 variables, i.e. four variables (AEP, PEP, $\ddot{\theta}_B$, $\dot{\theta}_B$) for each of the four legs (MIL, MOL, HIL, HOL). Subsequently, variations on the actual trajectory are analysed, even when the patterns of leg influence are similar. For this model, trajectories can be greatly affected by having a particular thoracic segment contributing more than the other segment, hence the right balance of parameter values is critical.

In the following sections, comments are made on results of varying active and passive parameters. Details of particular values can be found in appendix C.2.

5.4.1 Active parameters

Preliminary tests showed that the first condition is for both parameters to be greater than zero, that is, the turns cannot be successfully produced using only rotation of the mesothorax or only sideways movement of the metathorax. Hence, for this set of experiments, variables κ^{γ} and κ^{α} were given the following values: $\kappa = \{1.0, 1.3, 1.5\}$.



(a) Relatively dominant mesothorax (b) Relatively weak mesothorax (c) Mesothorax highly subordinated

Figure 5.11: Unbalanced mesothorax and metathorax behaviour; trajectories

Left: Mesothorax sideways movement overcoming turning in the metathorax. Middle: Weak mesothorax behaviour; trajectories are mostly smooth. Right: Large values for subordination or active control; both thoraces out of control.

Over this range only 2 out of 16 variable values cannot be matched to those in Figure 3.14. Varying the parameters affected only certain legs at specific locations, specifically, the influence of the MIL at the AEP, and the influence of the MOL and HIL at the PEP. Other variables like the average angular acceleration $\ddot{\theta}_B$ and velocity $\dot{\theta}_B$ were not affected as much. However, varying the active parameters does have a notable effect on the whole trajectory (Figure 5.11). Trajectories were less affected by the absolute value of parameters, but by the difference between them. It appears that active lateral movement produced by the mesothorax must be balanced with rotation generated by the metathorax. If κ^{γ} is high relative to κ^{α}, the mesothorax lateral force cannot be opposed by the hind legs and the body starts to move laterally. Consequently, unless the target is really far away, the turn almost never finishes. This type of trajectory is shown on the left hand side on Figure 5.11. On the zoomed area shown on the top right corner, one can see that body positions are almost parallel. On the other hand, if κ^{α} is high and κ^{γ} is low (or zero) the result is smoothly curving trajectories such as that shown on the middle plot on Figure 5.11.

In general, it was found that trajectories were more similar to the insect when $\kappa^{\gamma} > \kappa^{\alpha}$ with $\kappa^{\alpha} \approx 1.25$. Having both parameters with high values increases the probability of having instabilities, however instabilities were also sometimes observed in the stick insect behaviour under these conditions.

5.4.2 Passive parameters

Leg contribution and body trajectories are also affected by the subordination parameters introduced in section 4.2.1.1.

Initial analyses focused on varying the alpha joint subordinations s^{α} (which mostly affect forward speed) on both segments. Subsequently, the gamma joint subordination s^{γ} is varied. Although it is possible to vary subordination at the meso- and meta-thorax differently, the analysis was simplified by having the same value for segments. The set of subordination values tested was s^{α} or $s^{\gamma} = \{0.1, 0.3, 0.5\}$.

The first important fact is that variations in the alpha joint subordination have more effect on body angular acceleration than direct activation of κ^{α} or κ^{γ}. This indicates that it is very important to tolerate and follow external forces when trajectories are not explicitly calculated. However, the coupled forces created by middle and hind legs create a torque that would normally be cancelled by front legs. Consequently, trajectories become increasingly unstable with increasing values of the subordination parameters (Figure 5.11(c)). Insect trajectories with front tarsi blocked did oscillate more, but walking stability was not compromised.

After varying s^{α} on both the meso- and meta- thoracic segments, it was found that no single combination of parameter values for subordination in the alpha joints will produce a match for all 16 leg influence measurements, but in general higher values provided a better match. Only two of the insect's swing statistics (HOL at the AEP and average $\ddot{\theta}_B$ of the MIL) could not be matched. Lateral movement on the mesothorax is improved if the forward speed of that leg is reduced while turning; and this requires a reasonably large s^{α}_{meso} as there is no direct control of the alpha joint in this segment. Otherwise smooth trajectories are produced, like that depicted in Figure 5.11(b). On the other hand, the active control of the alpha joint in the hind legs will be negated by larger values of s^{α}_{meta}, producing an imbalance similar to that shown in Figure 5.11(a). However, when s^{α}_{meta} is too low, the leg contributions of hind legs is opposite to that of the insect, i.e. HIL contributes to turn whilst the HOL oppose turn.

Most of the leg contributions are unaffected by changing the value of the subordination parameters s^{γ}, particularly s^{γ}_{meta}. Nonetheless, setting both at maximum again tended to cause unstable turns. Turning is helped by reasonably strong subordination of the gamma joint in the hind legs, as this joint is not actively controlled and becomes important for forward speed as the leg reaches a rearward position, just before the PEP. As the value of s^{γ}_{meso} decreases, the MOL contributes less to turning at the PEP,

whereas the MIL contributes more at the AEP; this is consistent with the idea that a lower subordination increases effects by actively controlled joints. Despite this result, trajectories were improved by having some degree of subordination on the mesothorax gamma joints because it controls excessive lateral movements this segment might have.

5.4.3 Final results with front tarsi blocked

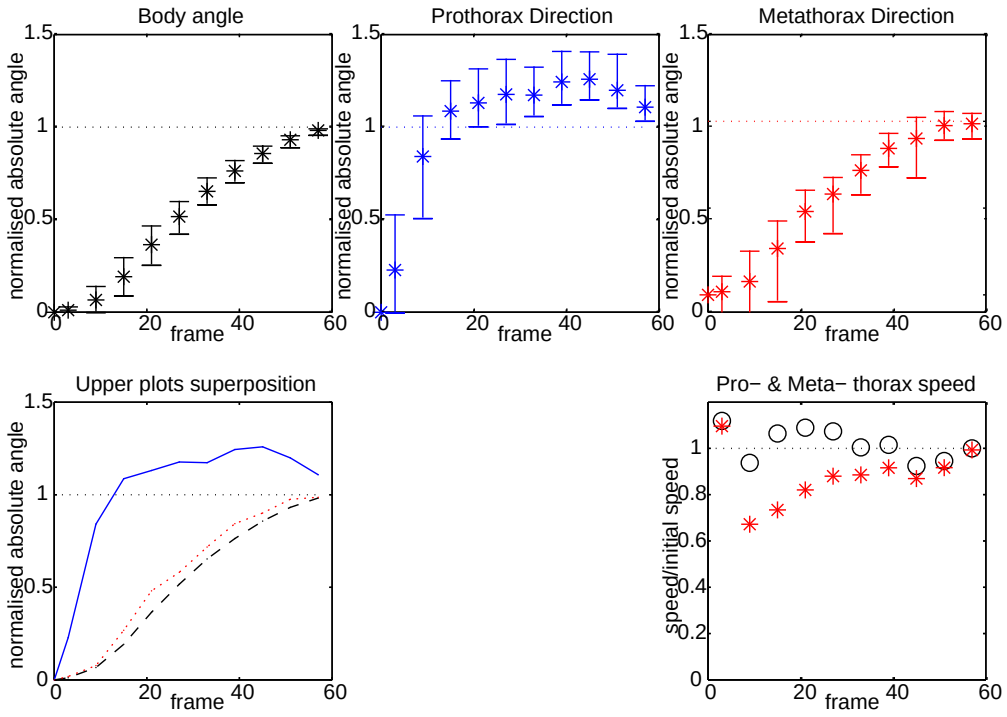


Figure 5.12: Body kinematics: Robot model when front tarsi are blocked

Absolute angles normalized to the target angle $\theta_T = 1$ (unitless) and the bars indicate standard deviation. Top left shows progress of the body angle θ_B, top middle is that of the prothorax θ_P and the top right is the metathorax direction θ_M. Superposition of top plots is shown in the bottom left; metathorax θ_M with a dotted line; and the body θ_B with a dashed line. The bottom right shows changes in speed for the prothorax ($\circ$) and metathorax ($*$) relative to their velocity before the turn. cf. Figure 3.12 for insect.

Based on the results from exploring the parameter values, the following final set of parameters for the model was chosen: $\kappa^{\alpha} = 1.25$, $\kappa^{\gamma} = 1.5$, $s^{\gamma}_{meso} = 0.3$, $s^{\gamma}_{meta} = 0.2$, $s^{\alpha}_{meso} = 0.3$, $s^{\alpha}_{meta} = 0.5$. Figure 5.12 shows that the direction of the protho-

rax θ_P and metathorax θ_M follow a similar pattern to the insect when front tarsi are blocked (Figure 3.12). Figure 5.14(a) shows that the AAR is also positioned between the mesothorax and metathorax, near to where the inner legs contact the ground. Simulation turns were characterized by alternations between short bursts of forward walking and rotation only; as seen in Figure 5.14(b). This qualitative result was also observed in the insect, however at the moment it is not possible to directly compare this behaviour because the insect exhibits substantial hesitation in making turns in these conditions (i.e. with front tarsi blocked), whereas the simulation turns quite consistently. The lack of normal ground contact may influence the animal in ways other than just the direct consequences on walking that were modelled here. Figure 5.13 shows the angular statistics of the simulation when front leg tarsi are blocked. The legs seem to play similar roles as with the insect, except for the effect of the HOL during its AEP and the corresponding body angular acceleration $\ddot{\theta}_B$ while this leg is swinging.

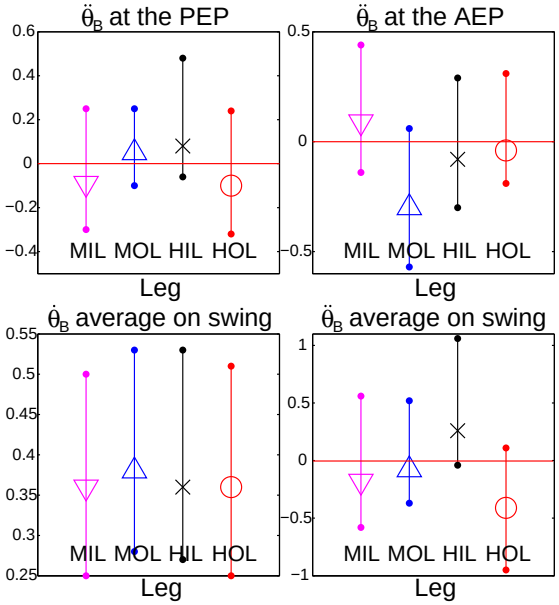


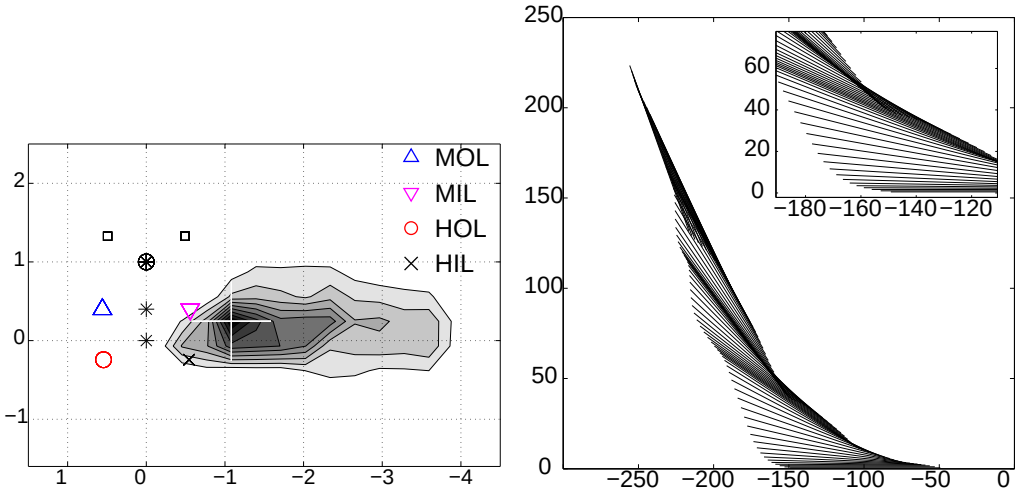
Figure 5.13: Swing statistics: Robot model when front tarsi are blocked

5.5 Final model

Finally, the complete system under normal conditions can now be tested, i.e. with the front tarsi unblocked, and the results compared to those for the simpler model in 5.2[5].

[5]Snapshots of this experiment can be found in Figure D.2.

Note that now the front legs are also partially influenced (through non-zero subordination parameters, $s^{\alpha}_{pro} = 0.25$ and $s^{\alpha}_{pro} = 0.15$) by the hind legs, to obtain complementary behaviour. Figure 5.16 shows that the simulation now responds more quickly to the target stimuli and holds the direction of the prothorax steadier (top, centre). The behaviour of the metathorax is more similar to the body (bottom, left) indicating that sideways movement is reduced. Overshooting the target angle sooner means that angular speed is now increased thanks to the metathorax and mesothorax active role. The prothorax direction, particularly the initial response, is smoother because front legs now respond to external forces. The front legs point more directly towards the target than is seen in the insect, although like the insect, the outer leg has larger values than the inner leg (bottom, centre). The metathorax speed is reduced (bottom, right) although still not as much as in the insect. Furthermore, the AAR is moved to fall at the same lateral distance as the hind legs; this is shown in Figure 5.15.



(a) Accumulative IAR (b) Typical turn

Figure 5.14: IAR: Robot model when front tarsi are blocked

Cumulative plot of the instant axes of rotation (IAR). The contours represent the distribution of the IARs, the axes is the distance normalised to the distance from prothorax to metathorax. The white cross shows the average axis of rotation (AAR). The three black asterisks (∗) represent the three thoracic segments and the prothorax (front) is encircled. All turns were properly transformed to the right before statistics were calculated. Middle ‘M’, hind ‘H’, inner ‘I’ and outer ’O’ legs ‘L’.

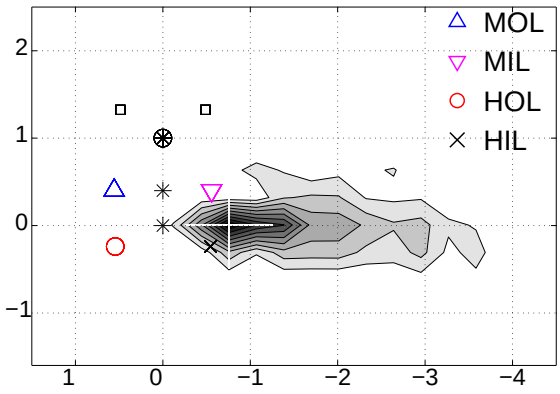


Figure 5.15: Robotic simulation turning with complete differentiation

Cumulative plot of the instant axes of rotation (IAR). The contours represent the distribution of the IARs, the axes is the distance normalised to the distance from prothorax to metathorax. The white cross shows the average axis of rotation (AAR). The three black asterisks (∗) represent the three thoracic segments and the prothorax (front) is encircled. All turns were properly transformed to the right before statistics were calculated. Middle 'M', hind 'H', inner 'I' and outer 'O' legs 'L'.

5.6 Discussion

This chapter presented a thoracic differentiation approach for the control of turning. This required adjustment to known coordination rules because some variables and parameters needed to be compatible. Furthermore, robustness was increased in order to minimise as much as possible problems related to leg coordination. Given the role the prothorax has for turning, a model where only this segment is actively inducing turning while the meso- and metathorax follow passively was presented. It was observed that trajectories have dissimilarities with those seen in the insect. Therefore, simple mechanisms were introduced on the meso- and metathorax that would aid with the rotation of the body.

In this chapter it was demonstrated that undifferentiated control of each segment does not produce insect-like turning results and that insect behaviour is better described when a complete thoracic differentiation is implemented. Calibration of the simulation was different for each thoracic segment, but was identical for either side of the robot. Nonetheless, leg trajectories on both sides differ significantly, just as with the insect. Each section in this chapter is now discussed in more detail.

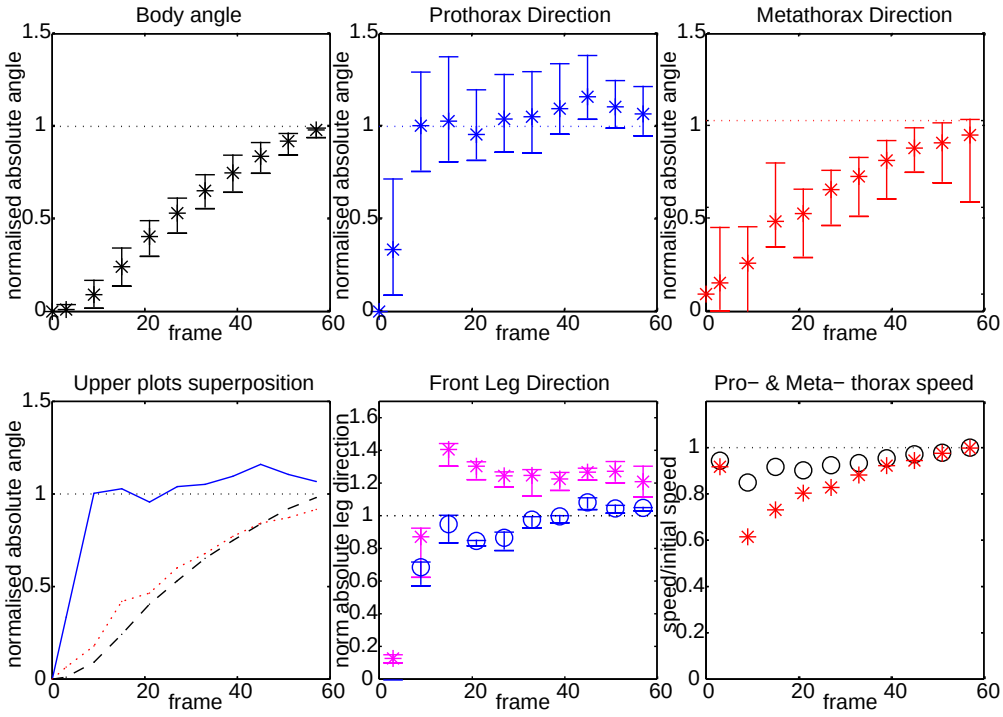


Figure 5.16: Body kinematics: Robot model with complete differentiation

Absolute angles normalized to the target angle $\theta_T = 1$ (unitless) and the bars indicate standard deviation. Top left shows progress of the body angle θ_B, top middle is that of the prothorax θ_P and the top right is the metathorax direction θ_M. Superposition of top plots is shown in the bottom left; metathorax θ_M with a dotted line; and the body θ_B with a dashed line. The bottom right shows changes in speed for the prothorax ($\circ$) and metathorax ($*$) relative to their velocity before the turn.The bottom middle shows the direction front legs follow relative to the initial heading, i.e., $\theta_B + \theta_L$; both legs are shown, inner front leg ($\circ$) and outer ($*$).

5.6.1 Leg Coordination

Leg coordination rules were successfully implemented and failure rate was very low. It was mentioned earlier that coordination rules are reportedly variable during turning (Dürr, 2005). For instance, according to rule 2 caudal position on the anterior leg induces stance-swing transition. However, the hind inner leg tends to be arrested for longer on tight turns, therefore, this rule could be weakened to induce this. Figure 5.17 shows the time legs spent on stance for various target angle positions. Note that the hind inner leg increases its time on stance not by decreasing effectiveness of rule 2, but by mechanisms of rule 5b.

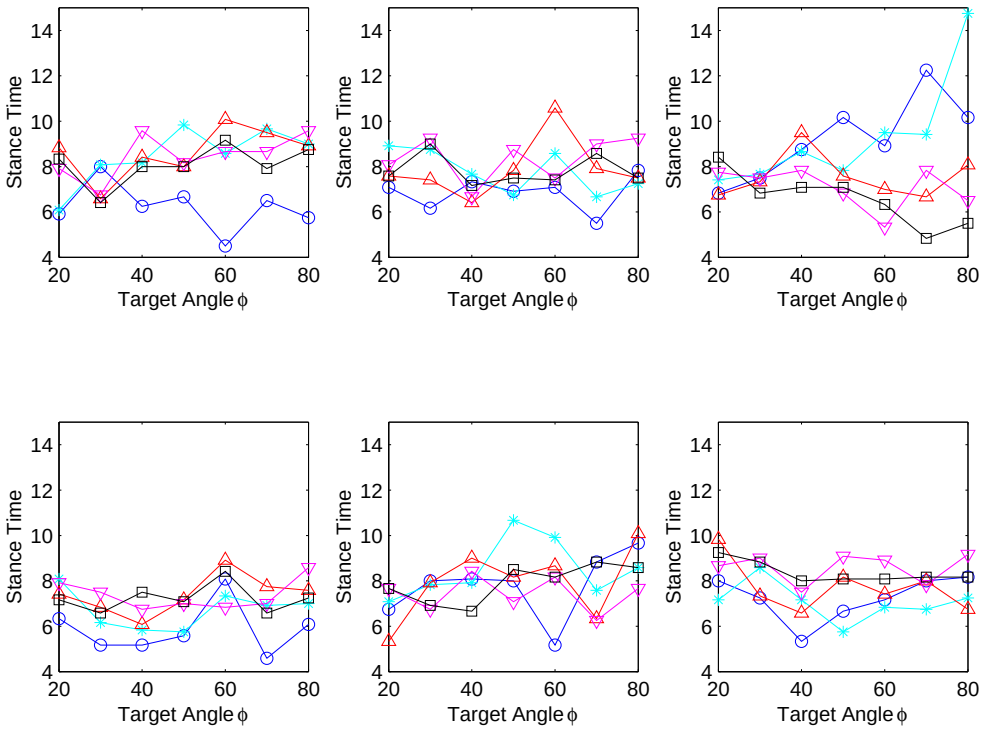


Figure 5.17: Average time on stance as the angle to the target varied

Leg coordination was not changed for any run. Upper row are inner legs to the turn, the prothorax is to the left and metathorax to the right. The number of step after the turn began is shown by the following symbols: first step ○ (blue), second ∗ (cyan), third △ (red), fourth (magenta), fifth □ (black)

It is possible to relax or remove some of the rules as presented in section 5.1. The suggestion that parameters controlling walking vary depending on the current behaviour can increase robustness. However, it is not possible with the present work to

support one approach over another, because the simulation does not implement variable parameters for leg coordination. Nonetheless, it should be noted that variable parameters were not necessary to obtain insect-like behaviour.

Additionally, it was found that coordination rules extend to all leg trajectories, as shown in Figure 5.18. This particular trajectory solves the navigational problem presented in section 1.1. A wheeled vehicle cannot replicate this motion.

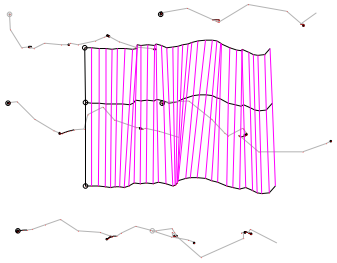


Figure 5.18: Lateral walking

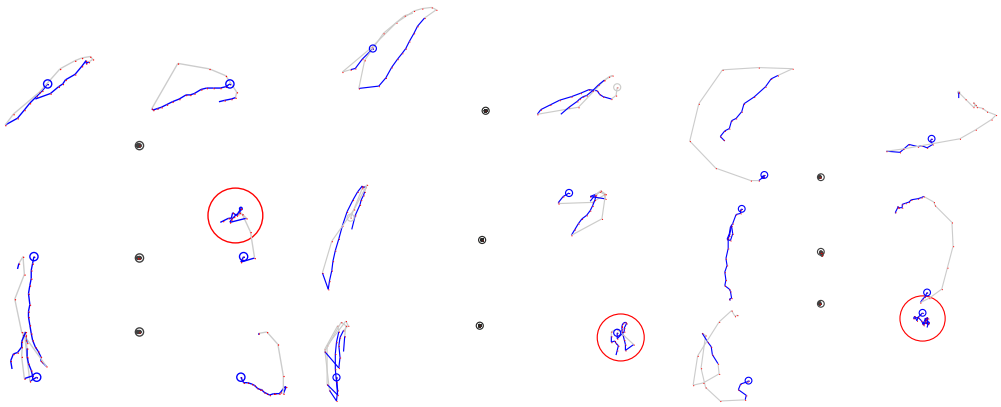
Leg coordination extends to all leg trajectories, even those not normally done by the insect.

5.6.2 Thoracic Differentiation

The front legs are important for turning and hence strongly influence the other legs through the subordination parameter. However they are known to be less important during forward walking on a horizontal plane, and for the model controlled by the prothorax only (section 5.2) two values of the parameters were needed to model different sizes of turns. A plausible improvement to the model would be to have these parameters modulated in a continuous fashion (Dürr and Matheson, 2003) according to the desired direction, with the influence of the front legs consequently reduced when the insect is not turning. Another change that might create a closer match to the exact leg trajectories seen in the insect would be to alter the speed of the metathorax or mesothorax legs during the turn; for simplicity, no speed changes other than those occurring through subordination were implemented.

Subordination values are highly dependent on the dynamics of the system. For instance, in the insect the metathorax is not at one extreme of the body, thus the inertia of the body relative to the mesothorax is lower. However, that extra segment increases inertia eight times. Therefore, subordination values are expected to be different for

each implementation. Nonetheless, the mechanisms that control turning are most likely constant, i.e., prothorax controlling target angle to zero, mesothorax pulling sideways mostly and metathorax rotating.



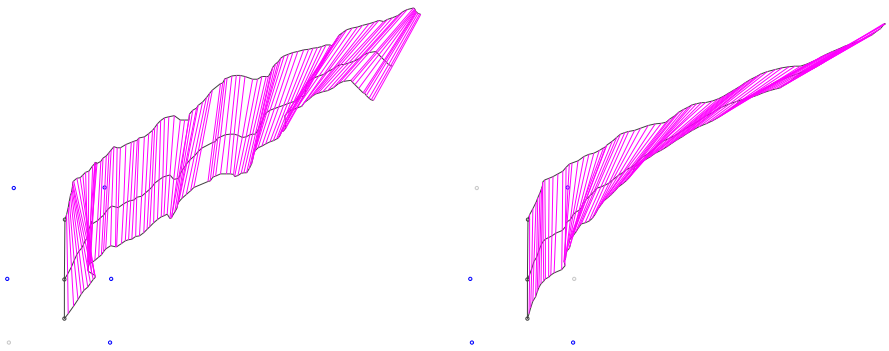
(a) Just Prothorax (b) Thoracic Differentiation (c) Insect

Figure 5.19: Comparison between simulation leg trajectories and an insect

Leg trajectories relative to the body show only the first third of the turn in all cases. Dark regions indicate stance and light is swing. Circles (red) highlight the leg that tend to be arrested during turning.

Calibration of the simulation was different for each thoracic segment, but was identical for either side of the robot. Regardless, leg trajectories on both sides differ significantly, just as with the insect. For example, the behaviour of the hind inner leg can replicate that of the real insect by becoming arrested in one position, without any explicit implementation of a stop. In Figure 5.19 typical leg trajectories are shown for the simulation and the insect. Without complete differentiation, the middle inner leg tended to be arrested. Based on the analysis presented in section 5.2.1 it could be possible to move the AAR further back, however, that would require increasing subordination in the mesothorax.

The subordination values set for the mesothorax and metathorax when these segments do not contribute to turn are already too high and prone to be excessively influenced by external forces. Due to time constraints, it was not possible to properly test how turns are influenced by external forces, e.g., tilting the surface. However, preliminary results, shown in 5.20, indicate that subordination values need to be restricted to a certain maximum in order to avoid deviations from gravity or alternative external



(a) Just Prothorax (b) Thoracic Differentiation

Figure 5.20: Targeting at 60 degrees, surface roll of 20 degrees

Left: The surface is rolled at 20 degrees. Because of the high subordination required on the middle and hind legs, the legs follow external forces too much. Right: Complete differentiation allows lower values of subordination on middle and hind legs, thus, targeting is not different from flat surfaces.

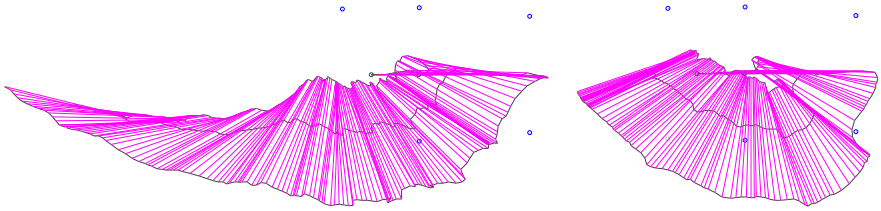
forces.

Experiments with the front tarsi blocked in section 3.6 showed that, when necessary, insects can turn on the spot using only four legs. However, it was noticed while recording real insects, that occasionally they willingly lift both front legs when faced with tight turns. Future work could examine using a combined strategy for making difficult turns. For example, a 180-degree turn could be easily achieved by first turning on the spot mainly with the middle and hind legs and then using front legs to finish the turn.

Initial tests for 180 degree turns indicate that the controller as it stands can turn by that much with considerable shortening of the distance to the target[6].

Turning control in the cockroach differs from results presented herein for the stick insect, mainly in the single leg roles at running speed (Jindrich and Full, 1999b). The cockroach's inner legs change the torque impulse so as to oppose turning; whereas the outer legs contribute to turning. Results for the cockroach report that its centre of mass (COM) direction is close to the body direction; this corresponds to a point close to the metathorax. According to results presented herein, the stick insect also has its COM heading similar to the body. However, further comparisons were not possible, because

[6] These are preliminary results, snapshots can be found in appendix D.



(a) Target at 8 m distance (b) Target at 4 m distance

Figure 5.21: Targeting at 180 degrees, distance variable

Left: The target is behind the robot, the distance is the same as with experiments presented throughout this section (8m). Right: Same angle, but now the distance to the target is half as with the example on the left (4m).

it is the prothorax direction that changes the most, therefore, no direct comparison between body trajectories could be estimated. Leg roles have more similarities at lower speed turns (Mu and Ritzmann, 2005). Nonetheless, based on different morphologies, these insects most probably have different single leg controller strategies for turning.

Chapter 6

Replacing the Jacobian by a Distributed Artificial Neural Network

The control architecture of the robotic model presented in chapter 4 was designed to replicate that of the real insect. However, the complete model for the stick insect is not known at present and a number of hypotheses were proposed (section 4.2). In order to facilitate comparison between the robot model and the insect, each control strategy was proposed with the real insect's capabilities in mind. For instance, no global variables were used, only those relative to the model and its own internal sensory information. However, it was assumed that the control of joint angular velocity could be computed by the insect, most probably in the thoracic ganglia. For this solution the analytical equation given by equation 6.3 was used (copied from 4.1), because it allowed us to focus on the interaction between joints. In this chapter, the possibility of substituting this equation with a biologically plausible neural network is tested. Based on results presented previously, it was found that it is not necessary to have adaptation or learning mechanisms to solve the stance phase. Furthermore, in section 2.4 it was mentioned that most non-spiking neurons are more related to the control of motoneurons. Therefore, it is assumed that it is not compulsory to substitute this equation by means of spiking neurons.

$$\dot{\vec{W}}_{sp} = \begin{bmatrix} \frac{\vec{T}_x}{\delta\alpha} & \frac{\vec{T}_x}{\delta\beta} & \frac{\vec{T}_x}{\delta\gamma} \\ \frac{\vec{T}_y}{\delta\alpha} & \frac{\vec{T}_y}{\delta\beta} & \frac{\vec{T}_y}{\delta\gamma} \\ \frac{\vec{T}_z}{\delta\alpha} & \frac{\vec{T}_z}{\delta\beta} & \frac{\vec{T}_z}{\delta\gamma} \end{bmatrix}^{-1} \left| \begin{array}{c} r\cos\theta_{\vec{L}} \\ r\sin\theta_{\vec{L}} \\ dz(h_{\vec{L}}) \end{array} \right| \tag{6.1}$$

$$= [J(\vec{T}(\alpha,\beta,\gamma))]^{-1}\vec{L}(r_{\vec{L}},\theta_{\vec{L}},dz_{\vec{L}}) \tag{6.2}$$

$$= \quad [\dot{\alpha}, \dot{\beta}, \dot{\gamma}]^T \tag{6.3}$$

The task is to calculate joint angular velocities $\{\dot{\alpha}, \dot{\beta}, \dot{\gamma}\}$ given the leg current position $\{\alpha, \beta, \gamma\}$ and the intended leg velocity $\vec{L} = [\theta_{\vec{L}}, r_{\vec{L}}]$ while controlling its height $dz(h_{\vec{L}})$. Of these variables, only $\vec{L}$ is assumed to be received from a higher-level, i.e. it is the control variable. It is easier to solve the problem in Cartesian coordinates without requiring global positioning, i.e., $dx(\theta, r_L)$ and $dy(\theta, r_L)$. Therefore, just as with the analytical solution, the input variables are $\{\alpha, \beta, \gamma, dx, dy, dz\}$ and the output variables are the three joint angular velocities. Note that the Cartesian representation refers to the vectorial base and not to the position. The transformation does not require kinematic transformation and thus it is straightforward to change bases.

In section 6.1 it is explained how data was generated for training and testing networks. Using this dataset, different networks of diverse characteristics and complexities are tested. In section 6.2 it is explained how these networks were trained and some preliminary results are presented. In section 6.3 different network topologies and key features are compared that support the artificial neural network replacing the analytical equation.

6.1 Generating Training Data

The first objective was to create a training dataset containing positions the leg could reach under 'normal' circumstances. In order to avoid bias, it was decided not to use data from the ODE simulation. Substitution of the analytical equation should also solve for future robotic implementations and different tasks, i.e. the alternative neural network should replace the control of stance not only for the range of movements used so far in this thesis. Furthermore, the model so far did not require swing target calculations and as a result the AEP did not change much for each step. Moreover, it was noted that stance trajectories were shorter than those created by the real insect, even if legs were capable of moving further. These characteristics were considered not to be necessary in demonstrating the operation of stance; however, they could induce even more bias on the network training.

Nevertheless, although the aim is for more flexibility and range of movement than that given by the ODE simulation, it would be meaningless to solve for all possible combinations of input variables. Hence, data was generated not by the dynamic ODE simulation, but by a kinematic model programmed in Matlab®. This model produced

controlled stepping with similar characteristics to the dynamic simulation, but with larger range of movement. Leg dimensions and capabilities were identical to those in the dynamic model. However, for this leg model it was necessary to compute an appropriate AEP according to a given leg direction $\theta_{\vec{L}}$. Otherwise, the same complications seen in the dynamic simulation could have emerged. This is described in the following sections.

Approximately 1,000 steps were generated by arbitrary leg directions $\theta_{\vec{L}}$ starting at an optimal AEP*. Conditions for local height and control on the $\hat{z}$ axis were also as before. These steps used equation 6.3 to compute leg joint velocities and samples of input-output data were taken every 10 time-steps. The total size of the dataset depended upon the distance legs could travel, the number of steps and their position along the body; approximately 25,000 data points for each joint. The input to the network was a vector with six inputs: three for the current position and three for the intended direction.

6.1.1 Optimal AEP

An optimal AEP* can be found such that the stance duration is maximised for a particular direction of step. Swing trajectories of the insect are mainly controlled by the CT (β) and BC (α) joints. The former moves the leg up and down, while the latter moves the leg towards the front (section 2.2). Normally the γ joint was moved quickly to a fixed position until touching the ground. The final position of the α joint varies for larger values of $\theta_{\vec{L}}$, whereas β always compensated for the distance from tarsus to ground. Therefore, it was needed to find the optimal α and γ joint positions for a given $\theta_{\vec{L}}$; β could be found based on the leg local height and the other joints once calculated.

Approximately[1] 10,000 steps were generated starting at an arbitrary position and at a random direction $\theta_{\vec{L}}$. In addition, the local height was randomly moved ± 0.1 and the leg vertical direction was specified based on average values used by the dynamic simulation $dz \approx 0.0008$. For each step the leg was moved in direction $+\theta_{\vec{L}}$ until the optimal posterior extreme position was found (PEP*), then it was moved in direction $-\theta_{\vec{L}}$ to find the maximum anterior extreme position (AEP*). From this dataset, the maximum stance distance from AEP* to PEP* was found for a given direction interval $\theta_{\vec{L}} \pm \Delta$. This was repeated for each thoracic pair of legs according to their own range of movements. Top left plot on Figures 6.1, E.1 and E.2 shows these results for the

[1] Some positions did not have a solution for β; most had two solutions for γ

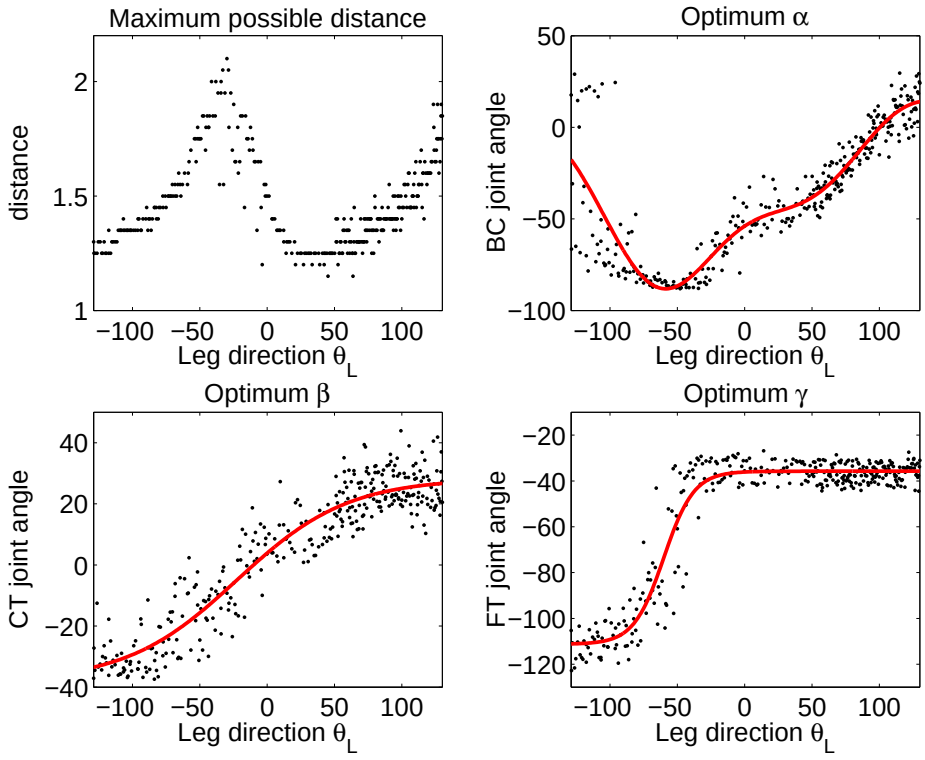


Figure 6.1: Prothorax optimal AEP angles fitting

Positive leg direction angles pull the leg towards the body; negative values away. Angles are shown in degrees. Top left: At a given leg direction range $\theta_{\vec{L}} + \Delta$ the maximum distance the leg could travel was found. Top right: Optimal angle for the α joint, this function is the most complex of all joints. Bottom Right: Optimal value for the γ joint, it basically only takes two extreme values. Bottom Left: Optimal value for the β joint. Although this joint has an optimal value, it is used to control local height.

prothorax, mesothorax and metathorax respectively.

The top right and bottom right plots on Figure 6.1 show the α and γ joints associated to the optimal AEP* that gave the maximum distance[2]. The function followed by the γ joint resembles a sigmoid function for all thoracic segments. The femur-tibia joint γ, contracts as much as possible for future negative directions (movement of tarsus away from the body), whereas for positive directions it is extended as much as possible (tarsus pulled towards the body). The body-coxa joint α final position is more variable, particularly when the range of movements is larger, e.g. as for the prothorax. From

[2]The mesothorax and metathorax results are in E.1 and E.2

these results, it can be seen that the β joint also has an optimal value (Bottom left plot on Figure 6.1), which means that there is also an optimal local height for each leg direction. However, the local distance to the ground for each leg is expected to be variable, furthermore, it could be counterproductive to have α and γ joints relatively fixed if β is variable.

6.1.2 Validating Optimal Angle Joints

For a given leg direction the optimal AEP* can be estimated. However, at each step there are vertical variations that result in a different AEP. In this section, the variation of the AEP is analysed when the ground is touched at different heights.

At the beginning of every step, it was assumed that each leg receives information about the direction $\theta_{\vec{L}}$ it must follow. However, although all legs control for a specific local height, there is no expectation for the touchdown position (AEP). This variability was solely implemented based on the β joint. Two artificial neural networks were trained to match functions of the expected AEP, one for the α joint at touchdown, $\alpha_{AEP}(\theta_{\vec{L}})$, and another for the expected γ joint touchdown value, $\gamma_{AEP}(\theta_{\vec{L}})$. Then a variable error was introduced to the $\beta \pm \varepsilon_\beta$ joint to estimate the error in tarsi positioning compared to the optimal. The maximum value was $\varepsilon_\beta = 20^\circ$.

Figure 6.2 shows variations in the AEP caused by changing only the β angle (optimal AEP position is shown by a solid line)[3]. The error varied according to the thoracic segment and the intended leg direction $\theta_{\vec{L}}$, however, in general the top-view positioning of the AEP was not compromised. The deviation from the optimum is only clear in $\hat{z}$. This indicates that it is possible for the leg to touch the ground at different heights while minimizing deviations on the 'x-y' AEP. This is a positive feature because legs depend upon the correlation between $\dot{\beta}$ and dz to find the ground. There were particular regions in which this approach did not allow for large variations in local height. Nonetheless, the range of movements and leg directions were sufficient to produce a reliable database.

6.2 Training the Artificial Neural Network

Although a traditional approach was used to compute error during training, the performance of all networks was tested by generating random steps. This decision was

[3]The mesothorax and metathorax results are shown in Figures E.3 and E.4

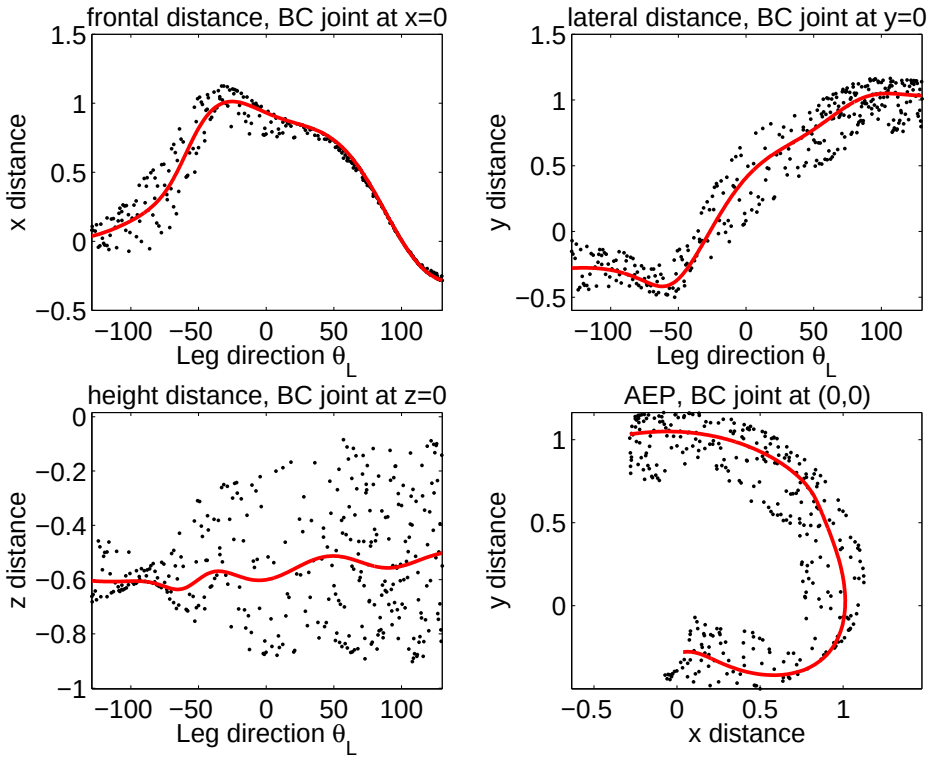


Figure 6.2: Prothorax optimal AEP with variations in the CT joint

Positive leg direction angles pull the leg towards the body; negative values away. Angles are shown in degrees. The coxa is at position $[0, 0, 0]$. Optimal values are shown by a solid line, dots represent variation due to β. Top left: positive values of x points to the front. Top right: positive values of y points away from the body. Bottom right: top view of the AEP. Note that β does not have a strong effect. Bottom left: Variation in the height of the AEP when β is moved by ± 20.

taken to avoid biases by networks fitting only the training dataset. Furthermore, it was necessary to estimate what would be the effect of implementing certain networks on the dynamic model. The equation to be replaced was highly non-linear, and as the size of the networks increased, alternative topologies and network sizes were considered and compared.

6.2.1 Inefficient Approaches

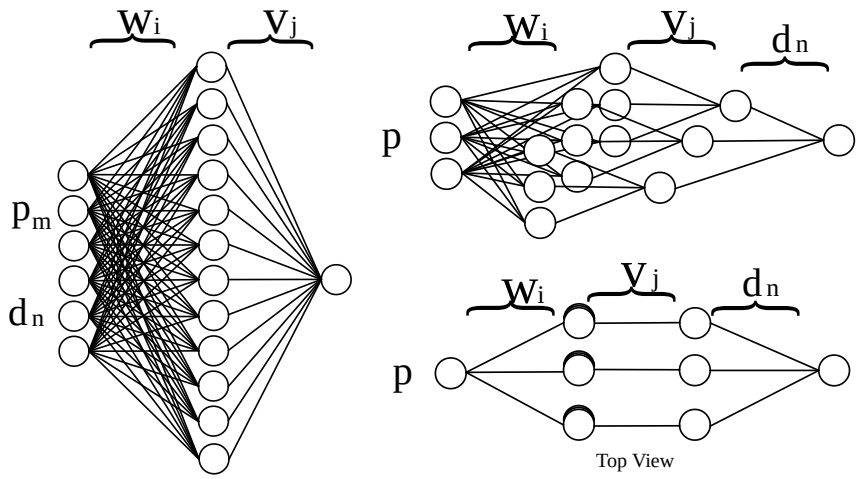
Initial tests with linear models proved to be insufficient to replicate the target equation, even for a very small range of joint movements. Therefore, it was decided to use

a multilayered perceptron with one hidden layer, as is shown in Figure 6.3(a). For all tests, two activation functions were tested: the sigmoid function and radial basis function. Initially, only a solution for the prothorax configuration was sought because it is more demanding in terms of variability, i.e., non-linearity. The training algorithm was run with the number of hidden neurons increasing from two to eighteen. However, trajectories created by using these trained networks showed considerable deviations. Testing the dynamic simulation resulted in severely impeded walking. Note that the interaction between legs in a dynamic model should compensate for some degree of trajectory deviation. Instead of continually increasing the number of interneurons or hidden layers, a different network topology was considered.

6.2.2 Dividing Input Processing

A different approach based on how the information was processed by the analytical equation was proposed instead of continuing increasing the size of the networks. The left hand side of the solution presented in equation 6.3 depends only on the position, whereas it is only on the right hand side that the intended direction is introduced. In other words, the non-linearity is entirely attributed to the inverse Jacobian at that position; thereafter, the relationship is linear. In addition, each of the nine terms that are computed for the Jacobian do not hold any relationship to the others. This observations suggests two important facts: one is that by splitting early stages of the network, approximations of the non-linear response could be easily achieved; and secondly, the intended direction should be introduced after the activation functions.

The proposed network topology is shown in Figure 6.3(b). The inputs controlling the intended direction are now the last weights. The three last interneurons represent the derivative of joint velocity with respect to each intended direction $d\vec{L} = [dx, dy, dz]$. In Figure 6.3(b) it can be seen that each of these processes information independently. If the non-linearity of the inverse Jacobian is well reproduced by the first layers it should solve for all possible values of $d\vec{L}$ because the relation is linear. For the topology shown in Figure 6.3(a) the range of values intended for $d\vec{L}$ would normally have to be introduced in training, limiting its performance.



(a) Typical network topology (b) Distributed network topology

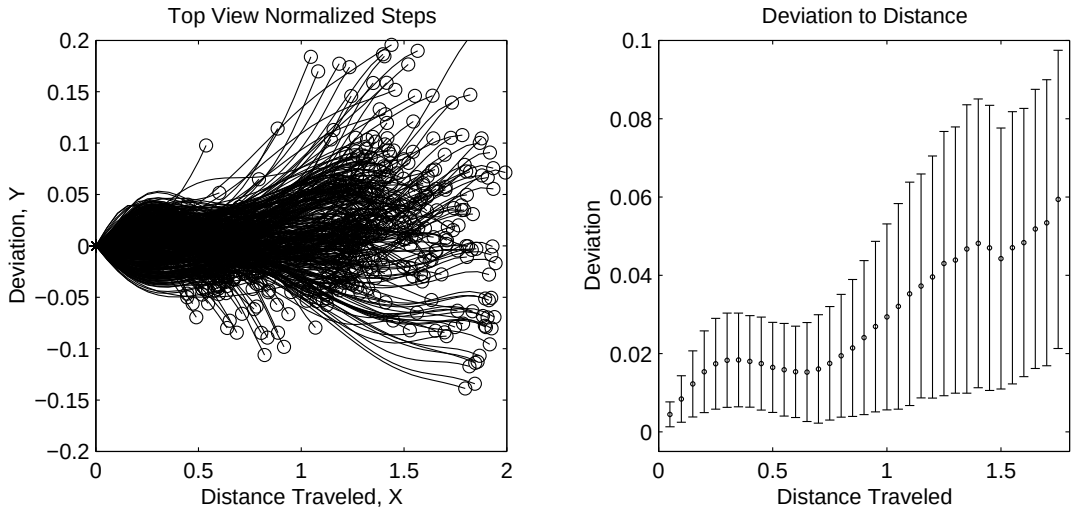
Figure 6.3: Artificial neural networks topologies tested

Left: Typical artificial neural network with one hidden layer consisting of 12 interneurons. All input variables are positioned to the left and connect to all interneurons. Right: This network topology has only three input variables to the left, the other three are now weights of the last hidden layer. On the top view it can be seen that there are three independent pathways that process the position, each of these correspond to directions of the intended leg direction $\{dx, dy, dz\}$. The total number of interneurons in this example is 12.

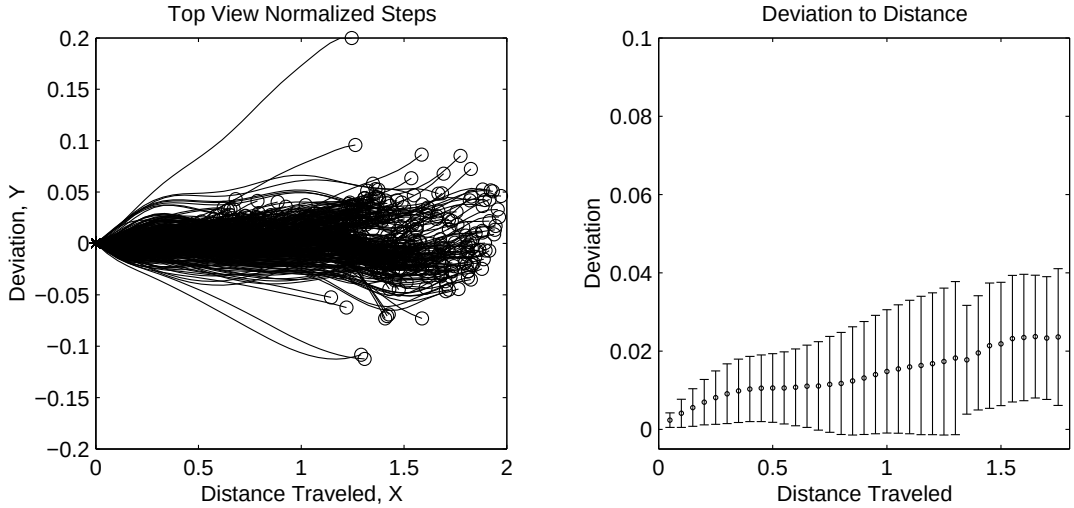
6.3 Comparing Network Topologies

Networks were trained with the same algorithm, implementing a conjugate gradient descent algorithm to minimize error; for each training step all data was used (batch mode). Network performance was not evaluated based on the fitting error, but on deviations produced by 500 random steps. Test leg steps were normalized to zero degrees. Figure 6.4(a) shows normalized steps done with the distributed topology using 12 interneurons; in Figure 6.4(b) the same topology but with 18 interneurons is shown. The latter network size represents deviation errors that when implemented in the simulation, indicate similar behaviour to that of the Jacobian. Leg trajectories obtained by the network with 12 interneurons work for forward walking and smooth turns, but not for tight turns.

It was tested how fast the two different topologies converged to a local optimum.



(a) size=12



(b) size=18

Figure 6.4: Direction normalised steps using proposed neural network

Top: Distributed network topology, total size=12. Bottom: Distributed network topology, total size=18. Left: 500 steps normalized at the same direction and same AEP at (0,0); the PEP is shown by (○). Right: mean and standard deviation of steps shown on the left. Only deviation with at least 50 deviation samples are shown. The error increases with the distance because it is accumulative.

The number of interneurons was set to 12 for both topologies and training was stopped after 10, 50, 100 and 400 steps. The left hand side of Figure 6.5 shows the distributed topology converges faster. Particularly at the beginning, results are improved by as much as half compared to the traditional topology. Although the performance of the distributed topology was better at all training steps, the traditional topology reaches similar values once it has stabilized in a local optimum.

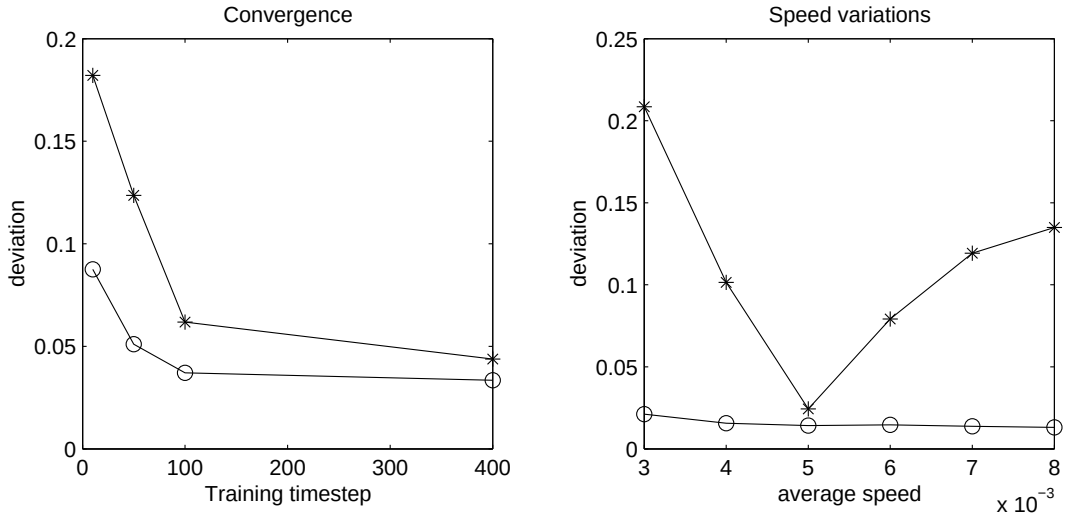


Figure 6.5: ANN performance comparison between tested topologies

Traditional network topology is shown by the solid line marked with (*); the distributed network topology is marked with (o). Left: Deviation resulting after training was stopped at 4 different stages. Right: When the average speed is moved away from that used for training the performance of the traditional network deteriorates.

It is important to test how these different topologies respond to a broad range of control values. The average speed for all experiments in the dynamic and kinematic simulation was set to 0.005 m/s. However, the average speed changes passively when turning, and should also be actively changed without compromising walking. It was mentioned above that the linear dependency present in the distributed topology should not be affected by variation in speed. For this experiment, 18 interneurons were used for both topologies and training ran for 1000 steps; this corresponds to a sufficiently stable network. Then 500 steps with variable average speed were generated. On the right hand side of Figure 6.5 it can be seen that the traditional topology was severely

biased to solve only for the particular average speed on which it was trained[4], whereas the distributed topology generalized for all speeds. This suggests that the distributed topology has the internal model represented in the first layer and the brain intention is controlled with the second layer. The traditional topology could be trained for a larger range of average speeds; however, this would reduce its performance. Furthermore, it is difficult to predict what the desired average speed would be. Essentially, the traditional approach learns the internal dynamics only partially, depending on the given commands given during training. It is therefore more advantageous to use a topology that learns the dynamic model itself.

6.3.1 Varying Network Size

The minimum number of interneurons necessary to substitute the analytical equation is analysed in this section. Figure 6.6 shows the performance of the distributed network with various number of interneurons. The network topology shown in Figure 6.3(b) is used for each of the joints in the leg, $\{\dot{\alpha}, \dot{\beta}, \dot{\gamma}\}$. First, the size of this network was changed equally in all three. The relationship between the size of the network and the deviation shows an inversely proportional tendency. However, local optima have a strong effect on the performance of network sizes, particularly for the metathorax. Small networks or less accurate networks, predominantly affect the prothorax; this is because the range of movement in FTs is more demanding.

Different networks of different sizes controlling joint angular velocity were combined. The leg controller used three networks for controlling each joint, one of these was set to one of these network sizes: $\{20, 24, 28\}$. The other two joint networks were reduced either by -3 or -6. In total 36 leg controllers of different sizes were tested, some of these were of the same total size but distributed differently. Figure 6.6 shows with the solid line the performance of leg controllers with joint controllers of equal size. Results do not favour having a particular joint controller with more interneurons than others.

6.3.2 Conclusions

This chapter demonstrates that it is possible to replace the inverse Jacobian matrix of the tarsus position used for the control of the single leg. It was also shown that tradi-

[4] Although average speed is specified during training, variations in speed do occur; the value specified is only the mean.

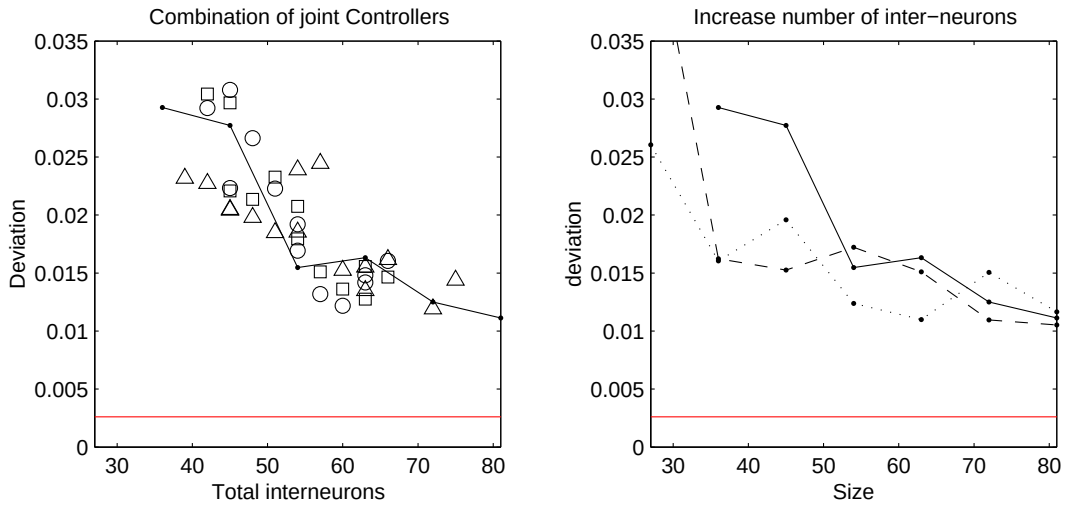


Figure 6.6: Performance of distributed ANN for variations in size

Left: 36 different combinations of joint networks for the single leg controller. Solid line shows symmetric sizes for all joints. Combinations with larger network sizes controlling the α joint are shown with (△); those with larger networks for the β are shown with (□); and with (○) those favouring the γ joint. Right: Performance of the distributed network topology varying total network size. The prothorax is shown with a solid line; the mesothorax with a dashed line; and the metathorax with a dotted line. Deviation with the analytical equation is shown with a continuous line at around 0.002.

tional artificial neural network topologies are less efficient than the topology proposed here. This is because traditional topologies require the input command at the same input level as the leg position. However, as seen in equation 6.3, each command input depends only on three of the inverse matrix elements. Furthermore, the dependency after the inverse matrix is linear.

Results indicate that only the total number of neurons in each leg varies the leg performance. However, this was only verified for the prothorax, which require more flexibility than the rearlegs. Future work could investigate if the same performance happens for legs that require less stance directionality.

The AEP used for generating the data is optimal only for the kinematical model. When this was implemented in the robot simulation, legs extend too much and could not support or pull the body efficiently. However, if the range of movement in joints is reduced it is possible to keep the AEP closer to the BC joint.

Chapter 7

Discussion

This thesis is based on two seemingly orthogonal fields, robotics and biology. The best solutions for locomotion are still found in biological systems, yet, relatively little is known about them. The philosophy that guided this research was that it is more beneficial to improve robots in parallel and based on current understanding of biological systems. Biorobotics is a win-win strategy because for a complete understanding of neuroscience it is necessary to implement an embodied numerical method for analysis, namely a biorobot. Similarly, if a robot could behave like an animal, most probably that system will outperform alternative robots.

This chapter concludes this thesis and presents avenues for future research. First, a summary of previous chapters is included and then the conclusion of this research is presented. After that, the more suitable next steps for this work are presented.

7.1 Summary of the Proposed Walking Controller

The target system, the *Carausius morosus*, was properly introduced in chapter 2. One key feature of the stick insect's nervous system is that the ventral nerve cord has three thoracic ganglia, each of which has its own walking generator. Additionally, each ganglion is partially split into left and right, morphologically and physiologically. These, and behavioural experiments, indicate that walking in the stick insect is decentralised. Walking in the insect is based on six decentralised, fairly independent leg controllers. The implications of leg individuality are enormous. Legs, contrary to wheels, need to 'reset' their motion, mainly when these become mechanically constrained, yet, transition from stance to swing cannot be stochastic. Results suggest that legs share some limited information allowing them to coordinate step phase transitions.

However, shared information between legs is limited and apparently sufficient for leg coordination.

Signals from the brain modulate walking, however, there is no dedicated information to co-ordinate all joints. Legs are mechanically coupled when on the ground. One question, yet to be fully answered, is how legs, and more specifically joints, move in such a way as to synchronise with the rest of the legs in the body. This is a non-trivial problem because the insect has 18 joints. A mechanism called the 'reversal reflex' based on (Bässler, 1976) suggests that joints follow external forces rather than oppose them, i.e., joints are under positive feedback. If the external force acted on the body in the direction of heading, all joints would eventually move in the direction they ought to move. No co-ordination would be needed between legs or joints within legs, instead, it would exploit the physics of the world. A dynamic analysis, however, shows a pitfall in this approach based on one simple fact; not all external forces correspond to the desired motion. On the other hand, Bartling and Schmitz (2000) demonstrate that joints always respond with a negative feedback to external disturbances. Additionally, in chapter 3 it was shown that legs also oppose inter-leg forces continuously; suggesting that negative feedback is always present. This thesis suggests that the reversal reflex is a parallel mechanism that aids coordination but does not dictate legs internal intention. Controlling directly the initial intention via negative feedback has some clear advantages; the most important being precise control and the ability to cope with external interferences. The distinction between external forces and inter-leg forces need not be made.

The most important contribution for the single leg controller was the integration of negative and positive feedback. It resembles concurrent work by Schneider et. al., however, an important difference of both approaches is how the reference for joints is calculated. In this thesis it is assumed that in each ganglia angular velocities for all joints can be calculated according to a particular tarsi direction. Individual joints are controlled with a cascade controller, which has the position in the last control loop under negative feedback, and the velocity controller, setting the new position, is modulated by the joint deviation. The position controller has the negative feedback mechanism whereas the velocity controller has positive feedback. This research demonstrates that the negative feedback is vital for the control of joints in dynamic systems. The forces involved in locomotion cannot be all used for ongoing motion. Initiation of movement and constant corrections of heading and speed require a negative feedback controller. Nonetheless, previous work that highlights the importance

of a positive feedback to coordinate decentralised single leg controllers has been supported. However, the controller proposed herein requires less information concerning intermediate states between motor command and segment position. It also allows for a more precise control of leg motion.

In previous chapters it was demonstrated that without the positive feedback, the system is partially defined as an open loop system setting joint velocity without any feedback, the servomotor or the musculoskeletal system is the actual negative feedback. The angle joint is defined by the resulting stiffness caused by a fixed motoneuron signal. In muscular systems, this is known as the equilibrium joint position (EP)[1].

Each joint controller receives information from other joints within the leg, and these are used to calculate the direction of the tarsus. Initially, an analytical solution that utilizes the inverse Jacobian of the tarsus position was used. However, it was demonstrated that this can be implemented using a distributed artificial neural network. Calculating the internal intention requires more computational power than controllers based solely on positive feedback. However, it is important to note that the system is still decentralised and controlling only three joints is still relatively easy.

Coordination of legs is not centrally controlled, but emerges from local rules. Nonetheless, these rules needed to be adjusted because some of the variables were not compatible with turning. For instance, Rule 1 was augmented by inhibiting contralateral legs as well; Rule 3 was partially based on distance travelled and not on caudal position; Rule 5b induced stance-swing when load on the leg was low. Leg coordination was not as stable as with the insect, however, it showed two of the most common gaits and an increased probability of favouring tripod gaits as the speed increased.

In order to analyse the complexity of possible behaviours with this decentralised architecture, it was decided to carry out alternative behavioural experiments with freely walking stick insects. The turning behaviour was chosen because it has the necessary elements for walking control. Commands from the brain are more likely simplified before getting to the thoracic ganglia because the brain is not essential for walking. Turning requires legs at every thorax to move differently, hence, it is possible to study the interaction between individual legs intention and cooperative mechanisms. The visual stimulus used in chapter 3 proved to be strongly correlated with front leg direction. This indicates that the signal is relatively precise (angle to target) and that legs have the necessary mechanism to respond accordingly. This experiment allowed the study

[1]These mechanisms are known to be present in various animals, including humans according to Kistemaker and Soest (2007).

of three important mechanisms of stick insect locomotion, the single leg controller, their thoracic differentiation and some of the signals that control legs. The approach is based on the fact that the IAR is not correlated to any part of the body or legs. Furthermore, it recognises that trajectory of legs and their speed is very likely to change while turning. Additionally, this thesis suggests that the COM need not be the more salient feature for a kinematic analysis.

After analysing the motion of stick insects towards visual targets during free walking, a new model for controlling leg movements during turns was proposed. The basis of the model presented herein is the decentralised Walknet proposed by Cruse et al. This thesis extends Walknet by introducing specific roles for each thoracic segment, which enables the robot simulation to accomplish complicated manoeuvres toward specific targets while maintaining the principle of a decentralized control architecture. It was demonstrated that undifferentiated control of each segment does not produce insect-like turning results and that insect behaviour is better described when a complete thoracic differentiation is implemented. The principal segment appears to be the prothorax, because it defines most of the body trajectory, differences in speed and the average axis of rotation. Controlling the turn by having just the front legs directed to the target and the other legs following passively produced reasonable results. However, the mesothorax and metathorax have active roles that complement the prothorax; and that can suffice to produce turns to the target even when the front legs are impeded. It was also demonstrated that without a thoracic differentiation turns do not replicate the biological target. This differentiation not only suggests how the insect walking control is organized, it also highlights a better strategy for turning in general. The important role of front legs relative to the middle and hind legs is vital because the proximity the former has to eyes simplifies possible calculations.

The simulation has the dynamic characteristics of a large robot, as demonstrated in chapter 4. It is also possible to scale the whole robot simulation, and preliminary qualitative tests show no difference in performance. All parameters used for the control of legs can be scaled linearly to the size of the robot and its weight. The only exceptions to this linearity are the parameters defining the damp-spring mechanisms at the joints. So far, these parameters need to be tuned heuristically, however, the range of values is very robust. Variations in weight can be as large as 16 times the original weight and the size of the robot can be changed as much as four times. Limitations are mostly related to ODE instabilities. Robustness in size and weight tested in section 4.5.3 indicates that the controller is a good candidate to succeed once implemented in a physical robot,

although this has yet to be confirmed. The robot simulation can be easily controlled by using the two global variables used: The target direction and the single leg speed. With these two simple variables it is possible to achieve complicated manoeuvres, for instance, those shown in section 5.6. Those trajectories, as mentioned in section 1 (Figure 1.2), are difficult to achieve by wheeled vehicles. Therefore, behavioural results achieved by the herein proposed robot simulation are attractive to implement in a real robot.

Some of the stick insect's locomotion resources for navigation were shown. Particularly motion on flat surfaces; nonetheless, path navigation was non-trivial. The model does not directly implement neuromorphological information because information is limited and biased to a particular leg. However, modules in the simulation could potentially be replaced by alternative processing units. Similar to the procedure followed for chapter 6, which replaced the inverse Jacobian with an alternative artificial neural network.

7.2 Future Research

The approach followed to propose a single leg controller was based on key aspects of leg behaviour and neural physiology. However, details about neuromorphology were not considered, i.e., how neurons controlling legs are interconnected. Nonetheless, there are still important gaps to cover before implementing a more realistic neural model for the control of legs. For instance, it was assumed that spiking neurons were most relevant for sensory filtering and information convergence, thus, temporal neural properties were not used. However, these are known to explain adaptations in leg trajectories during swing (Diederich et al., 2002; Schumm and Cruse, 2006) based on previous models of local bistable joint controllers (Cruse, 2002).

7.2.1 Single Leg Controller

The robot model presented herein utilizes two different algorithms to calculate the velocity on joints given the current position and the desired direction. The inverse Jacobian of the leg angular position and a distributed artificial neural network. It is plausible to implement this with a biological network, however, there are certainly different mechanisms that could potentially provide this information. The question of how inaccurate joint velocities can be is still open, because the fitness used in chapter

6 was based on a kinematic model. Trajectory error must be smaller than that reported by the kinematic model because all legs are mechanically coupled through the ground.

It was shown in previous chapters that it is possible to divert the direction of the meso- and meta- thorax by introducing a simple bias to one of the joints in either thoracic segment. This works mainly because the activation was applied to a known stable leg trajectory, i.e., tarsus motion was that of straight walking. An alternative solution for front legs, instead of solving for all tarsus directions, is to similarly bias just the solution for forward walking. The procedure would be to further distribute the ANN proposed in chapter 6 by solving only for straight walking and thereafter introduce a different ANN that would modulate that output for all directions. Indirectly, this has already been done for the mesothorax and metathorax because the angle was never set to any value other than zero. Therefore, the ANN with the angle set to zero can be simplified. After this network, additional terms with the activation factor can be included. Note that the tarsus direction will not be that of ϕ, but a direction that support rotating that much, e.g., lateral movement for mesothorax legs. Finding this 'simple' trick for the front legs is potentially the most useful step for future research.

7.2.1.1 Subordination

For most experiments, the subordination parameter had to be actively changed according to the task. It is clear, however, that resulting deviations are only detected at a higher-level by vision or similar sensory information. Implementing a variable subordination at a lower level is not appropriate because its effects are relevant at a trajectory level. It is possible to have a global subordination value that simultaneously affects all legs, thus avoiding having to control so many variables, which in any case are just six. This can be exploited by the brain for controlling behaviours that are more complex. For instance, it could set global subordination to zero when camouflaging or increase it as the terrain becomes more variable. In section 5.4 it was shown that subordination need not be a precise value. There, calibration was based on variation of parameters and function fitting, however, once the basic mechanisms are understood their calibration is relatively intuitive. Therefore, a higher-level mechanism could control these variables collectively according to the task. For instance, in the rolled surface experiment, joints more affected by gravity are the femur-tibias and in the pitched surface experiment were the body-coxa. Adjusting the latter by reducing the subordination in those joints would result in leg coordination rule 5, because a reduction in subordination would eventually be measured as an increase in power stroke. Additionally, future

work could address the issue of where subordination is modulated, the two candidates being the ganglia or the brain.

Previously, only the BC joint and the FT joint have been controlled with a positive feedback because the CT joint would eventually move the body towards the ground. Nonetheless, if the local height is known, it is possible to continuously control the height by increasing or decreasing vertical speed, i.e. $dz \neq 0$. Although it is advantageous for the CT joint to subordinate as little as possible $s^{\beta} \approx 0$, the possibility of requiring some subordination for climbing and rough terrain locomotion cannot be ignored. On the other hand, vertical subordination may be advantageous for walking on rough terrain. For instance, if one leg is relatively above the others, the motors will be under much stress, that would not happen it the legs adapt to some extent to different heights.

7.2.2 Leg coordination

Coordination rules were implemented based on simple rules that influence transitions between step phases (see section 5.1). Rules were modified in their implementation, however, no fundamental changes were made. Nonetheless, calibration by the GA resulted in unpredictable gaits within same trials. The reason is because the evaluation function was not sufficiently bounded or gait orientated, i.e., it was only based on distance and stability. Therefore, the simulation can travel large distances without falling, however, gaits are unstable and variable within the same conditions.

One possibility is that the increase in redundancy added to some rules unbound legs to change phases carelessly, i.e., the probability to swing was unnecessary high at times. Alternatively, because the simulation does not have tarsi, it is easier to lose contact with ground due to other legs. It is possible that rules based on load signals are more likely to induce transitions under this condition.

Although the resulting gaits were perhaps inefficient, known step gaits (tripod and tetrapod) proved not to be vital for a successful locomotion. Nonetheless, future work should pay more attention to coordination influences. The first step would be to remove redundancies and evaluate performance for all turning situations. It is clear that some rules need to be adjusted and others do not. For instance, rule 3 (caudal position excites start of stance on posterior leg) does not fit well on legs moving mostly sideways and rule 2 (start of stance excites start of stance on anterior leg) need not change to increase HIL time on stance.

It is not possible with the present work to support or refute active variation in coordination influences because the simulation does not implement variable parameters for leg coordination. Future work could analyse to what extent basic rules can be calibrated online to fulfil the same work.

7.2.3 Neurophysiology

Results presented in chapter 3 show that vision strongly influences thoracic behaviour. It should be possible to include similar visual stimuli in known physiological preparations to record how particular joints are affected. For instance, if a vertical black object is presented for the right eye and the insect is active or willing to walk, it should be possible to measure relative activity on flexor motoneurons in the middle inner leg. This is in order to verify one of the hypotheses presented in section 3.7.2.

Of similar interest, would be to record activity of the front inner leg motoneuron pools controlling the femur-tibia joint for visual input at various angles. For an open loop situation the response should be different if the tarsus direction is continuously controlled, whereas a monotonous activity would probably indicate a stereotyped reaction to turn at a non-specific angle, i.e., 'left' or 'right' turn, rather than turn at a θ angle.

If visual signals from the brain were simple as suggested by this research, it would be very insightful to find that signal as it gets to the ganglia. If replicated, it could allow a more precise study of locomotion because this higher-level signal modulates walking in a predictable way.

In general, this thesis supports the use of visual stimuli for neurophysiologal experiments following an analogous visual induced behaviour as that used in chapter 3, i.e., highly contrast vertical stimuli on white background. Note also that results could be biased for some preparations if vision input is not controlled.

7.2.4 Literature on target approach

The visual system is not fully analysed in this thesis. It is assumed that the angle to the target can be calculated for the body and leg trajectories. This model is a good platform to test eye models and brain processing in the stick insect because it is possible to compare directly the resulting effect of vision.

Qualitative observations led us to believe that the vertical target is approached using one eye. This is not evident when the target is at a large angle from the heading, as

it usually was to induce turning. In this situation, the target can only be seen with one eye in any case. However, when the object was in front, the vertical bar was not completely in front of the insect, but biased towards one side, i.e., it was towards the side of the first eye that saw the visual target. As the distance to the target became less, the angle to it would have considerably increased. Nonetheless, the stick insect kept walking forward without rotating. At about a couple of centimetres before reaching the objective, the stick insect made a final turn; at that distance the antennae had touched the target. Therefore, the stick insect seems to prefer tracking the object with one eye and it can identify when the target is in front. Then, rotation no longer corresponds to the angle to the target, it now seemed to wait for antennae contact. This behaviour requires further inspection. It does not contradict any of the assumptions presented in previous chapters. However, it indicates that the angle to the target is not the only signal controlling turning. Additionally, based on the position of the eyes, the stick insect has little or no stereo vision. Since the eyes are on either side it would be interesting to analyse how this affects targeting when the object is in front.

Experiments by Jander and Volk-Heinrichs (1970) presented two patterns at either side of the insect. The insect then decided which one to approach and thus it was possible to determine the most attractive shape. The experiment was stopped after a couple of steps because only the initial decision was important. It would be important to verify if the decision favouring one shape changes over longer periods. For instance, two vertical bars of different thickness might be different because one is bigger or because one is closer. Which one is favoured by the stick insect? Could it detect the difference?

How much more complex is vision? For instance, crossing branches might lead to two different treetops, one of them more attractive. Let us assume the one to the left is more attractive, but to get there it needs to take the right branch. Raw visual processing would lead the stick insect to walk to the left, a more ‘thoughtful’ approach would make it take the right branch.

7.2.5 Adaptability

Front tarsi were blocked for some experiments, however, these were blocked only temporarily because of the interest in studying middle and hind leg behaviour as if the front legs were still under normal operation. Turns were slower, the stick insect hesitated more and initially the body oscillated more. Now that a model mostly for

intact walking has been developed, the next step would be to prolong the time tarsi remain blocked and study how the insect adapts to this situation. If walking and turning adapts towards a more stable locomotion, it would be important to analyse what are the changes, because that would match more closely a robotic implementation. A more invasive experiment could analyse behavioural changes if the COM is moved close to the mesothorax and all tarsi are blocked. This should more closely relate to results presented by the robot simulation.

Adaptability was not considered for most experiments. Nonetheless, the question of whether stick insect became accustomed to experimental procedures done for this research is unknown. Insects were constantly reused for most of their lifespan, which is significantly longer than other invertebrates; living up to a year. Most repetitions were for preliminary tests, calibrations for the visual tracking, visual target selection (ultimately vertical bar), blocked tarsi tests and others. Now that the experimental procedure is well defined as stable, it would be ideal to reduce the frequency insects are reused. This would verify that the insects used for the behavioural experiments did not adapt in some way. Additionally, for experiments blocking the tarsi, it was intended to avoid insects adapting to walking under this condition. Nonetheless, it cannot be guaranteed that stick insects did not recognize this situation afterwards.

7.2.6 Implementation

The joint arrangement is based on a spring mechanism located between the motor's axis and the segment it is moving. However, it was shown that the controller proposed in chapter 4 only used the command sent to the motor and the segment position. Therefore, the spring mechanism can be replaced by adjusting the subordination parameter. Initial tests with the 2D limb dynamic simulation demonstrate this is possible, nonetheless, formal analysis needs to be done on the simulation. Future work can take a closer look at performance differences between both implementations. More importantly, future work could focus on the transfer of this controller to a real robot. Initially, it could be sufficient to add the subordination parameter to an existing joint controller.

Appendices

Appendix A

Visual Tracking Algorithm

A.1 Visual Transformation

A.1.1 Rotation

To calculate just the angle for the rotation matrix, the stance marks on each frame were shifted to its mean, (centre of mass)

$$\vec{M}_k^* = \vec{M}_k - \frac{1}{N}\sum_{i=1}^{N}\vec{M}_{k,i}. \tag{A.1}$$

The error between frames is given by the following equation.

$$E(\theta_k) = \sum_{i=1}^{N}(\vec{M}_{k,i}^* - R_k(\theta_k)\vec{M}_{k-1,i}^*)^2 \tag{A.2}$$

$$\frac{dE(\theta_f)}{d\theta_k} = 2\sum_{i=1}^{N}(\vec{M}_{k,i}^* - R_k\vec{M}_{k-1,i}^*)^T\Big(-\frac{dR_k}{d\theta_k}\vec{M}_{k-1,i}^*\Big) = 0 \tag{A.3}$$

The derivative of the rotation matrix is equivalent to the following expression,

$$\frac{dR_k}{d\theta_k} = R_k(\theta_k)R\Big(\frac{\pi}{2}\Big) \tag{A.4}$$

Furthermore, R is an orthogonal matrix, therefore, the inner product of vectors is zero due to the $\pi/2$ rotation.

$$(R_k\vec{M}_{k-1,i}^*)^T\Big(\frac{dR_k}{d\theta_k}\vec{M}_{k-1,i}^*\Big) = \tag{A.5}$$

$$(\vec{M}_{k-1,i}^*)^T R_k^T R_k R\Big(\frac{\pi}{2}\Big)\vec{M}_{k-1,i}^* = \tag{A.6}$$

$$(\vec{M}_{k-1,i}^*)^T R\Big(\frac{\pi}{2}\Big)\vec{M}_{k-1,i}^* = 0 \tag{A.7}$$

Equation A.3 becomes,

$$\sum_{i=1}^{N} (\vec{M}^*_{k,i})^T \left(\frac{dR_k}{d\theta_k}\right) \vec{M}^*_{k-1,i} \quad = \quad 0. \tag{A.8}$$

Expanding terms and regrouping $\cos\theta_k$ and $\sin\theta_k$, the following equation for the angle is found.

$$\cos\theta_k \sum_{i=1}^{N} \vec{M}^*_{k,i} \times \vec{M}^*_{k-1,i} + \sin\theta_k \sum_{i=1}^{N} \vec{M}^*_{k,i} \cdot \vec{M}^*_{k-1,i} = 0 \tag{A.9}$$

Using the tangent identity, $\tan\theta_k = \sin\theta_k / \cos\theta_k$ it is now straightforward to get final expression.

$$\theta_k = \arctan\left(\frac{\sum_{i=1}^{N} \vec{M}^*_{k,i} \times \vec{M}^*_{k-1,i}}{\sum_{i=1}^{N} \vec{M}^*_{k,i} \cdot \vec{M}^*_{k-1,i}}\right) \tag{A.10}$$

A.1.2 Translation

Based on the rotation angle given by equation A.10, the instant axis of rotation (IAR) is found. This is the position were a rotation θ_k of $\vec{M}_{k-1,i}$ equals $\vec{M}_{k,i}$. In the following equation the IAR at frame k is represented by $\vec{\Upsilon}_k$.

$$\vec{M}_k + \vec{\Upsilon}_k = R_k(\vec{M}_{k-1} + \vec{\Upsilon}_k) \tag{A.11}$$

The linearity with respect to $\vec{\Upsilon}_k$ makes it unnecessary to minimize for E^2, the error for the stance transition is given by the following equation,

$$E = \sum_{i=1}^{N} \left(R_k(\vec{M}_{k-1,i} + \vec{\Upsilon}_k) - \vec{M}_{k,i} - \vec{\Upsilon}_k\right) \quad = \quad 0 \tag{A.12}$$

$$\sum_{i=1}^{N} \left(R_k \vec{M}_{k-1,i} - \vec{M}_{k,i}\right) + \vec{\Upsilon}_k(R_k - I) \quad = \quad 0 \tag{A.13}$$

The final expression is given as follow,

$$\vec{\Upsilon}_k = (R_k - 1)^{-1} \sum_{i=1}^{N} (R_k \vec{M}_{k-1,i} - \vec{M}_{k,i}) \tag{A.14}$$

A.1.3 Scale

Small variations in scale are introduced by moving the camera close or away to the ground. The scale is calculated after the transformation matrix is calculated. The error is given by the following equation.

$$E = \sum_{i=1}^{N} \left(\vec{M}_{k,i} - \sigma_k \vec{M}_{k-1,i} \right)^2 \tag{A.15}$$

$$\frac{dE}{d\sigma_k} = -2 \sum_{i=1}^{N} \left(\vec{M}_{k,i} - \sigma_k \vec{M}_{k-1,i} \right) \vec{M}_{k-1,i} = 0 \tag{A.16}$$

The scale is given by the following equation,

$$\sigma_k \sum_{i=1}^{N} \vec{M}_{k-1,i} \cdot \vec{M}_{k-1,i} = \sum_{i=1}^{N} \vec{M}_{k,i} \cdot \vec{M}_{k,i} \tag{A.17}$$

$$\sigma_k = \frac{\sum_{i=1}^{N} \vec{M}_{k,i} \cdot \vec{M}_{k,i}}{\sum_{i=1}^{N} \vec{M}_{k-1,i} \cdot \vec{M}_{k-1,i}} \tag{A.18}$$

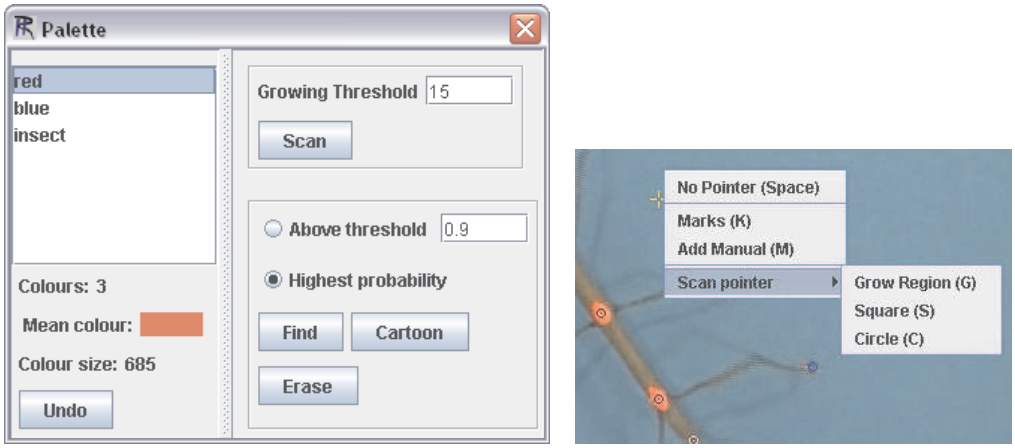
A.2 Colour classification

The tracking algorithm is based on image colour classification; it does not consider shape or temporal variables. Marks to be tracked are painted with a colour not in the scene. The colour to be tracked is found based on statistic of samples taken from that image and colours on the background. Therefore, each sequence would have at least two colour categories, the one to be tracked and the others are considered as background. All categories are specified by selecting sample colours directly from images, therefore, each category is represented by various pixel samples, which was called a compound colour. Once compound colours have sufficient samples, it is then possible to calculate the probability of a given sample belonging to each of these. The colour classifier finds all pixels of the tracked mark; this is then used to pinpoint the location of interest. Each of these stages are now explained in turn.

A.2.1 Image sampling

The user takes from different frames, preferably separated in time, samples that belong to individual compound colours. It is also important to consider the amount of samples each compound colour has, because that would be the probability a priori for finding

that colour. The number of compound colours depends on the similarity the tracking colour has to others on the scene. Most of the sequences analysed required between two and four compound colours. On Figure A.1(a) the graphical user interface for the



(a) Palette (b) Selection

Figure A.1: Colour Selection GUI

Palette and Main Canvas menu used to classify colours

colour selection is shown. There are currently three ways by which user selects an area of pixels to add to a given compound colour. Figure A.1(b) shows options available on the canvas, these can be a square or a circular region, or a growing region based on colour proximity to a reference colour. The reference is selected by the user, the algorithm then checks colour distance of neighbouring pixels. The colour distance between pixel 'n' and pixel 'm' is calculated in the following way.

$$cd = \sqrt{(r_n - r_m)^2 + (g_n - g_m)^2 + (b_n - b_m)^2} \tag{A.19}$$

A.2.2 Compound colour

Samples for compound colours are taken preferably from different sequence frames and from different positions. The density function to represent compound colours is the Gaussian function given in equation A.21. Pixels are represented as three-dimensional vectors; dimensions correspond to each of the RGB (red, blue, green) values, i.e. $\vec{x} = [r,g,b]'$. It is important to have diverse samples for compound colours because the density probability depends on Σ^{-1}. Very similar or few samples would make this

covariance matrix numerically unstable. Furthermore, within one frame it is not always possible to take many samples for one compound colour, for instance, on the tarsi there can be as few as 10 pixels.

$$\begin{aligned} p(x|\mu,\Sigma) &= \frac{1}{\sqrt{2\pi\Sigma}}\exp\Big(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\Big) \\ \mu &= \frac{1}{P}\sum_{i=1}^{P}x^i \\ \Sigma &= \frac{1}{P}\sum_{i=1}^{P}(x^i-\mu)(x^I-\mu)^T \end{aligned} \tag{A.20}$$

A.2.3 Classifying

The method used for classification, is density estimation with a Gaussian density function. Each of the K compound colours CC_n is represented by μ_n and Σ_n, $n = \{1 \ldots K\}$. The probability of finding a colour x^* given μ_n and Σ_n is given by equation A.21. Therefore, the probability of a class given a sample colour is found using Bayes rule, this is given by the following equation.

$$p(CC=n|x^*) = \frac{p(x^*|CC=n)p(CC=n)}{p(x^*)} \tag{A.21}$$

The probability $p(CC=n)$ (probability a priori) is based only on the amount of sample colours used to compute μ_n and Σ_n. Each compound colour used P_n sample colours, therefore, the probability a priori is given by the following equation.

$$p(CC=n) = \frac{P_n}{\sum_{i=1}^{K}P_{n=i}\,\forall i \neq n} \tag{A.22}$$

It is not necessary to calculate $p(x^*)$ when the algorithm finds the most probable class for a given sample colour, because it is only necessary to know the relative value of $p(CC=n|x^*)$.

A.2.4 Regrouping

The images used were compressed, usually in jpeg[1] format, resulting in dotted patterns around marks tracked. Therefore, regions formed by compound colours found by the

[1] Joint Photographic Experts Group

classifier are in fact many irregular segments of various sizes, the smallest ones often not connected. All pixels found within the scanned area were grouped together by pixels that were interconnected. The group with the largest amount of pixels was considered to be desired mark, the exact location being the average position of pixels in that group.

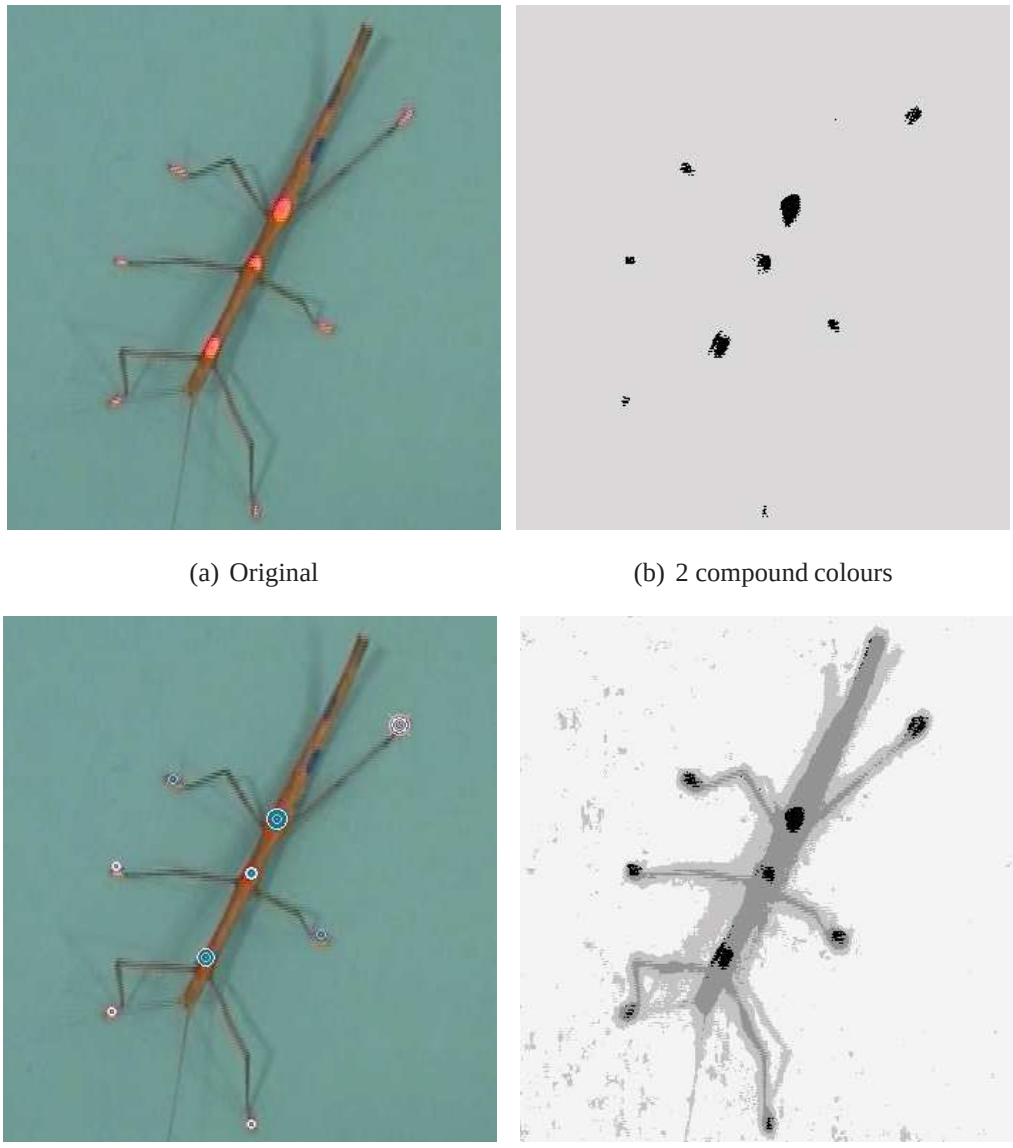
A.2.5 Results

Figure A.2 shows how the algorithm interprets a particular sequence frame. The top left figure shows the original image with an insect marked on nine places. To the right, top and bottom, two results from two different sets of compound colours are shown. The top right shows classification made with two compound colours, $K = 2$; the bottom right used four $K = 4$. Marks were red and the background was mostly light blue, insects were either brown- or green-like. Figure A.2(c) shows marks with a circle proportional to the number of pixels used to calculate the average of a given group.

2 compound colours		4 compound colours	
Red	508	Red	154
Background	21234	Background	4476
		Insect	3253
		Shadow	3457

Table A.1: Number of samples used for each compound colour. These colours were used to classify examples shown in Figure A.2.

The information contained on the image classified by four compound colours could in theory be used to improve automatic tracking even further. Legs are clearly visible and most importantly, they are differentiated from their shadows. This classifier can be used to track various objects provided colours are sufficiently conspicuous. For instance, this classifier was used in Lindsell (2005) to identify coloured circles on top of various robots. It can also be used to track the whole of the insect provided it appears sufficiently small to be considered as a point.



(a) Original (b) 2 compound colours

(c) Marks found (d) 4 compound colours

Figure A.2: Colour classification example

Top left: Original image, marks are painted red, background is homogeneous. Top right: Using two compound colours to classify original image. Bottom right: Classification using four compound colours. Bottom left: resulting mark points after classification.

A.3 Graphical User Interface

The colour selector and classifier, the mark-tracking algorithm and the analysis options were all integrated in the same graphical user interface (GUI) programmed in Java™. This section briefly describes how the GUI operates and what are the options the program has available.

The main window is the '*canvas window*', which shows the original images from the sequence. This window controls the mouse behaviour depending on the sequence analysis stage, i.e. if it is on the classifying, calibration or analysis stage. At the moment the program does not support direct video format as input, it must be provided in sequence of individual images.

A.3.1 Colour Sampling

The '*palette window*' shown in section A.2 takes samples from the '*canvas window*' and also displays pixels that have been selected. There are a number of options to control and evaluate colour calibration. It is possible to classify colour not by finding the most, but by finding that above certain confident threshold of belonging to that class; the selection criteria and threshold are shown on the palette window. Also it is possible to replace all pixels on the original image by the most likely compound colour on the palette. This creates a cartoon-like representation of the original image that is very useful to visually evaluate compound colours on the palette[2]. It was mentioned in section A.2 that selecting colours can be done by finding nearby colours close to a reference. The minimum distance to consider a colour sample close to the reference is specified by the 'Growing Threshold' parameter on the palette window.

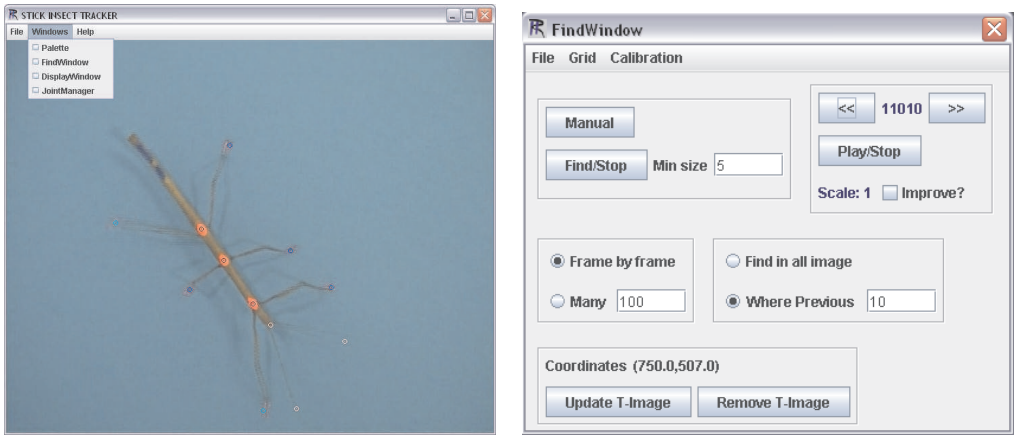
A.3.2 Tracking Marks

The windows shown in Figure A.3(b) controls the sequence flow and the algorithm to find marks; this is the '*find window*'. On the top right hand side button region there are options to move forward or backwards in time, as well as scaling the image and improving image contrast. Marks can be added manually or automatically by using the colour classifier; the algorithm can also ignore marks when less than a given number of pixels represented that mark[3]. These options are found on the button region on the

[2]Figures A.2(b) and A.2(d) used a greyscale instead of the compound colours on the palette for printing outputs.

[3]Refer to section 3.3 for details on the criteria to calculate position from a group of pixels.

top left corner. Other options control the pace used by the tracking algorithm, frame by frame, or batch processing. Also, it is possible to choose between finding marks on the whole of the image, or just where the previous marks were found. The scan area of the latter is also specified by the user; the default is 15 square pixels.



(a) Main canvas (b) Find window

Figure A.3: GUI for insect tracking

Left: The main windows is the canvas displaying the original sequence Right: This windows is mainly use to track marks on a sequence.

As explained in section A.2, mark positions are given by the average position of the largest group of pixels belonging to a given compound colour. The following step is to track the position of all marks in the sequence, each of these mark-sequences would represent a point of interest that might not be necessarily related to other mark-sequences. On the other hand, many of these could be tracking the same object but at different positions as to indicated orientation and well as position. Therefore, mark-sequences must be labelled accordingly in order to allow further processing. This information can then be used to describe motion of objects in a variety of different ways.

Although it is possible to detect marks on the whole of the image, this is not always necessary and it could be very time consuming for large images. Normally the number of marks does not change over time, and they are usually located close to where they were on the previous frame. Therefore, the tracking algorithm can find marks only near the previous position.

Additionally, various features were incorporated to compensate for errors in clas-

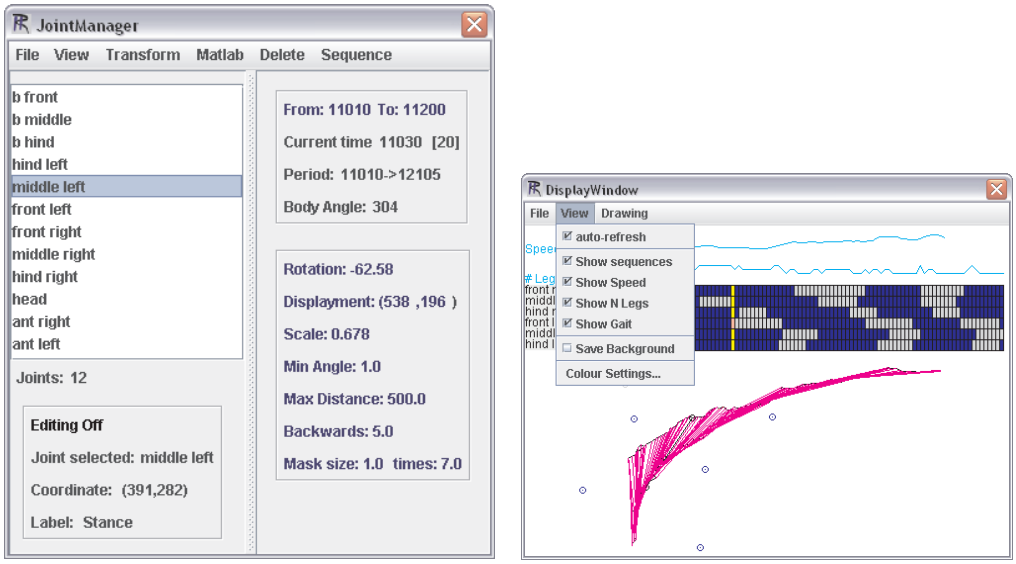
sification or to improve automatic positioning. Such as manually adding marks, reposition them or delete extra marks. Most problems whilst tracking stick insects occur when legs where swinging, because the slow recording rate was not sufficient to compensate for blurs or marks disappearing from the scanned region.

The tracking algorithm processes information only between adjacent frames, therefore, there are a number of problems that cannot be solved as it is. For instance, when two marks get close to each other both end up in the same place; the one with most pixels. It should be possible to use information about leg phases to estimate where marks are going to be, instead of just looking around the previous position. The main problem is to decide if step phases are to be labelled based on the velocity and position of marks, or if mark positions are to be found using step phase information. Alternatively, uncertainty by itself could mean that step phase changed.

A.3.3 Editing Mark-sequences

The '*canvas*', the '*palette*' and the '*find*' windows are mainly for taking the raw data of all marks during the sequence. There is no compensation for camera movement and there are no differences between the marks following legs and body. The '*joint manager*' window is used to process, manipulate and output sequences and the '*display window*' visually shows results as they are changed. These windows are shown on Figure A.4. The '*joint manager*' control the transformation of camera movement described in section 3.3. Mark-sequences are shown in the left list, shown on the left of Figure A.4(a). It is possible to edit in great detail each mark-sequence, from single mark position adjustment to combination of two mark-sequences.

Labelling mark-sequences are necessary for the rest of the algorithms to work properly. Labels were: *Body*, for the thoracic segments; *Stance* or *Swing* for each of the step phases; and the *Reference* label was just the default. The name was left to the user, and it is not considered for processing. Therefore, in order to distinguish between body segments, these have to be positioned orderly from front to hind. Labelling information is then used to identify adjacent stance marks in neighbouring frames or to identify the position and orientation of the body. Options like showing trajectories with respect to marks on stance or with respect to the body.



(a) Joint manager (b) Display window

Figure A.4: GUI for processing sequences

Left: The joint manager window is used to process and analyse sequences. Right: The display window is used to visually analyse data.

A.3.3.1 Lens Perspective

Depending on the experiment setting, mark-sequences described so far can be used as they are. For instance, if the camera does not move during for the whole of the sequence, and if lens distortion is minimal, one can use mark coordinates directly to describe motion. Unfortunately, at least one of the previous conditions cannot be fulfil satisfactorily. Below the proposed solutions implemented for compensating camera limitations on the tracking algorithm are presented.

Additionally, even if it is possible to compensate for camera complications, there is still an intrinsic problem with the way the tracking algorithm works. The setback is that marks in the image are not precisely one pixel size, therefore, there is a position variability when calculated mark position. The position variability, or noise, will depend on how big the mark, and how many of pixels representing the mark are actually found. A smoothing procedure for mark-sequences is available, it is done by averaging mark at time t with m marks before, the whole process is then repeated for a given number of times.

Camera movement is described in section 3.3.1, a different camera compensation

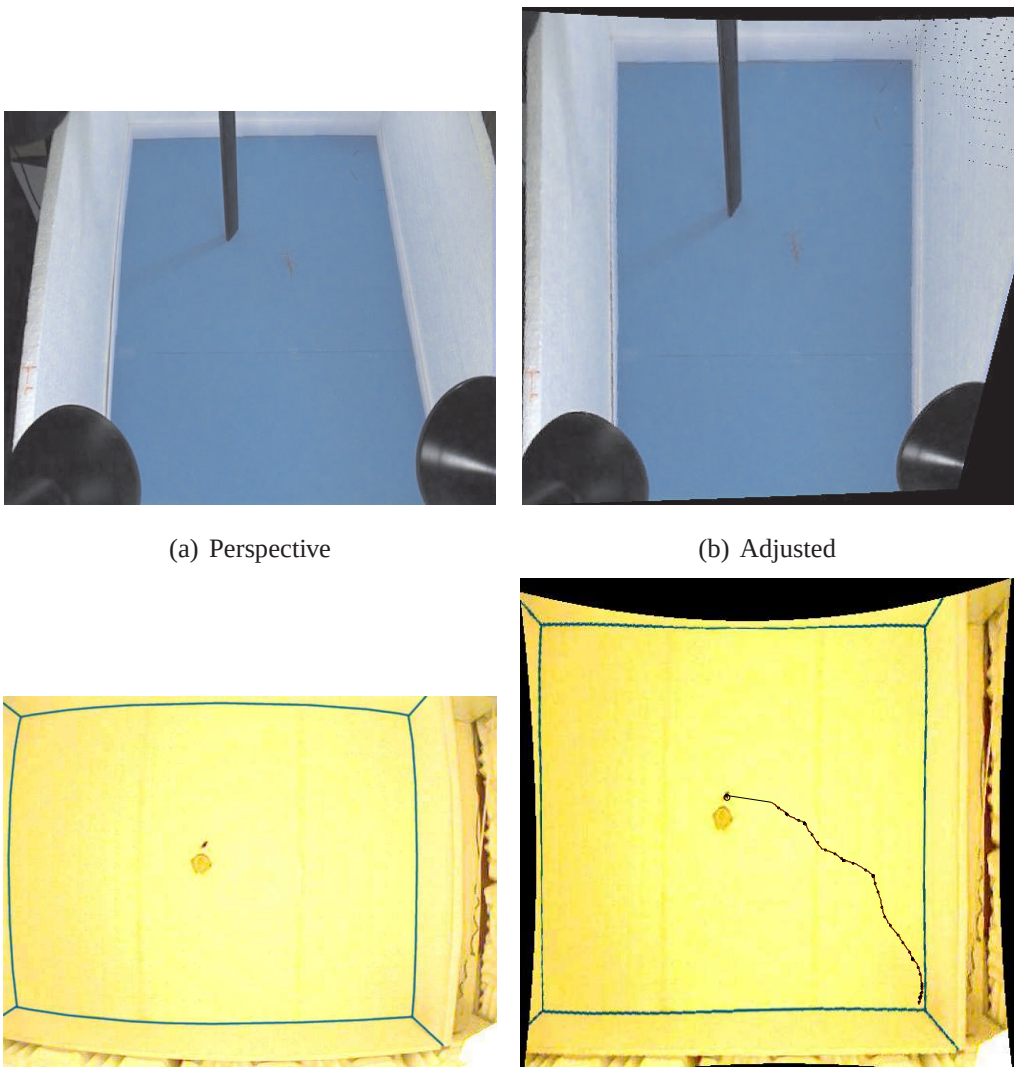
is described: lens distortion and perspective effects. The transformation assumes that at a given height y on the image, the transformation for x position can be done approximated by a quadratic function. Because this variation is different at each y, coefficients of this quadratic function are also represented by quadratic functions. This is described by the following equations.

$$x^* = x^2 a_1 + x a_2 + a_3 \tag{A.23}$$

$$a_i = y^2 k_{i,1} + y k_{i,2} + k_{i,3} \tag{A.24}$$

Therefore, at least nine positions have to be specified on the canvas and the real coordinates given in a file. Coefficients are found by calculating the minimum square error, therefore, transformation improves with more samples. In a similar way, parameters are found for y^* using the same data points.

Figure A.5 on the top images shows an aerial view of the arena before and after perspective correction. This camera calibration procedure was not used in the experiments presented in this thesis because insect recording were mostly like those shown in Figure A.2(a). The camera was not significantly tilted and the lens distortion at that distance was minimal. However, it has been used in other projects within the institute were the camera lens distortion was large. Figure A.5 on the bottom images shows an example of this program tracking a cricket from a top-view. Experiments shown for the cricket were kindly provided by research PhD student Mark Payne, who is currently studying visual-auditory integration in the cricket. The cricket was sufficiently small to be considered as a single mark, two compound-colours were used for classification, and the lens distortion was corrected as shown. It is possible to display the transformed image on the '*display window*', options are found at the bottom of '*find window*'. It is possible that before hand the area of interest is known, in this situation it is possible to predefine it. Figure A.6 shows that an arena that constrained insect motion within a circular area. For similar situations, it is recommended to select beforehand the area of interest, thus avoiding any possible misclassifications with the figures on the arena's wall. This is an image example provided by Dr. Jan Wessnitzer and research PhD student Michael Mangan, who are currently investigating spatial learning in the cricket.



(a) Perspective (b) Adjusted

(c) Lense distortion (d) Adjusted

Figure A.5: Camera compensation examples

Top left: Image distorted by the perspective of the scene. Top right: Image corrected by calibrating it with real coordinates. Bottom left: Image distorted by the camera lens. Bottom right: Image corrected by calibrating it with real coordinates.

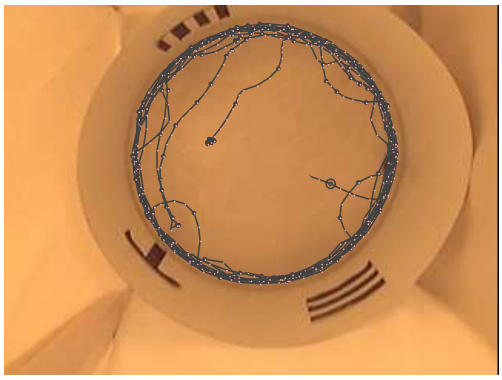


Figure A.6: Preselecting circular scanned area

A circular area corresponding to the shape of the arena was selected as the only area of interest. The algorithm ignores everything outside that area, thus simplifying the task.

Appendix B

Decentralised Walking Model

B.1 2D of freedom simulation

The two segments forming the limb are 0.2 and 0.15 metres long, with a homogeneous linear density of 0.8 kg/m, i.e. the whole limb weights, 280g. Each segment was considered to be of a cylindrical shape for calculating the moment of inertia. At the joints a damping mechanism was introduced in the simulation to introduce energy loss, and hence, realism. The external force was applied only horizontally on the tip of the second segment and gravity acted on both segments. The external force was normally set to $-0.3N\hat{x}$ and the target velocity was $0.05\hat{x}$

The servomotors were PD controllers, and their maximum torque was set to the same value for the whole simulation. If not specified otherwise, this maximum was set to $25kg \cdot cm$. However, the control parameters were optimised for each motor, the one on the base was set to $K_P = 4$ and $K_D = 0.1$ and the other to $K_P = 0.5$ and $K_D = 0.01$. Servomotors' angular set-points were updated every 20 ms, however, just as in a real servomotor, position was controlled continuously every 5 ms.

The velocity of the joints was calculated with the inverse Jacobian of the tip position. The tip position tp is given by the following equation,

$$tp = \begin{vmatrix} (\cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta))l2 + \cos(\alpha)l1 \\ (\sin(\alpha)\cos(\beta) + \cos(\alpha)\sin(\beta))l2 + \sin(\alpha)l1 \end{vmatrix} \tag{B.1}$$

Therefore, after derivation and combining using trigonometric identities, the Jacobian is given by the following equation,

$$J_{tp} = \begin{vmatrix} -\sin(\alpha+\beta)l2 - \sin(\alpha)l1 & -\sin(\alpha+\beta)l2 \\ \cos(\alpha+\beta)l2 + \cos(\alpha)l1 & \cos(\alpha+\beta)l2 \end{vmatrix} \tag{B.2}$$

Thus, the angular speed for each joint is given by the following equation,

$$\begin{vmatrix} \dot{\alpha} \\ \dot{\beta} \end{vmatrix} = J_{tp}^{-1} \begin{vmatrix} \dot{x} \\ \dot{y} \end{vmatrix} \tag{B.3}$$

B.2 ODE: World Simulation

The Open Dynamics Engine^{TM}(ODE) is a library for simulating rigid body dynamics. The version used for most experiments was 0.5, it is a very stable version with integrated collision detection and friction. These libraries are accompanied with a simple visual interface that outputs the same basic shapes used by ODE. The source code available was programmed in C++, therefore, the robot simulation and additional visual interfaces were also programmed in that language. The libraries contained basic predefined shapes that were used to specify the whole of the robot. The visual interface is just for visual evaluation, for most experiments and data gathering the simulation was run without display.

B.2.1 Environment

The two main component of the ODE simulation are the rigid body dynamics and the collision detection. The former is similar to particle dynamics with additional feature of considering mass and orientation, i.e. it considers inertia. However, it does not consider any deformation of bodies as soft-body dynamics engines. On the other hand, collision detection includes algorithms for calculating interaction with other objects, like impulse, friction and impacts. Computational simulations normally have to compromise accuracy for stability and computational time. Because of that, a number of parameters need to be set in order to control accuracy and interaction behaviour.

Some of the simplest simulation parameters are the gravity of the world, and definition of ground surface. These, along with the mass of the objects, specify the dynamics for the rigid body dynamics. However, most complications are related to the collision detection engine because the interaction between many bodies is by far the most complicated. Bodies are in theory rigid, but between simulation steps bodies can penetrate

others and bodies attached by joints can move away. 4.11 shows the environment created for testing the robot simulation.

B.2.2 ODE parameters

The error reduction parameter (ERP) controls by how much expected positions are corrected. Values ranges from zero correction, to one, which fully corrects in one time step. The ERP is particularly for controlling joints, however, contact between bodies are also influenced by the ERP. Setting the world ERP to zero would produce undetectable objects because the collision would not stop objects going through each other. Ideally, this value should be set to one, however, the deviation will not be corrected due to numerical approximations. Furthermore, the simulation will now be very unstable, particularly if the time step is not considerably reduced. Therefore, the collision world error in ODE needs to be present to avoid instabilities.

When two bodies are joined, the forces that the joint applies to the bodies are such that certain positions and directions are constrained. However, it is possible to modify that force by adding an additional term that would reduce its effectiveness. When the constraint force mixing (CFM) is zero, the force calculated is not modified, thus it is a hard constraint. Just as with the ERP, this would induce singularities in the calculations, and the simulation will be unstable.

Individual joints and contact points can be controlled by individual ERP and CFM. It is also possible to set these parameters in such a way that the constraint behaves more like a spring-damper system. ODE documentation reports the following equations to estimate spring k_p and damper k_d constants.

$$ERP = hk_p/(hk_p + k_d) \tag{B.4}$$

$$CFM = 1/(hk_p + k_d) \tag{B.5}$$

Frictional forces are approximated with the Coulomb friction, which is a force that opposes motion based on the relationship between normal and tangential forces. The coefficient of friction is normalised with the maximum force that can be applied at the contact point.

B.2.3 Robot Simulation Additional Features

The program itself allows for additional alterations of robot morphology that initially were used for testing individual control modules. For instance, it is also possible to remove pairs of legs leaving as few as two, when this happens, the rear is supported by a socket-attached ball. Additionally, it is possible to add an extension to the rear on the body, a tail. However, this moves the centre of mass to the back and then it would require tarsi on front legs. The body itself could be made stick insect-like, i.e. cylindrical and thin. In general, the robot simulation could mimic more closely the dynamics of the insect by changing the global scale of segments and by making the body cylindrical.

Appendix C

Thoracic Differentiation

C.1 Prothorax Pulling Rotation Model

Figure 5.6 shows a system of spring positioned in each rear segment, these represent opposing forces to the prothorax pulling, represented by F. This is a good model because each leg is trying to move forward and thus lateral displacement is negatively controlled. Furthermore, legs non-subordination means that the springs on Figure 5.6 follow the body. Therefore, a constant prothorax force would eventually result in a stable position for the springs.

Two spring connected on the positioned can be represented by a single spring whose spring-constant is the sum of the single springs. Therefore, the resulting opposing force by the mesothorax is represented by k_{meso} and metathorax force by k_{meta}. The stable position requires system total force and total torque to be null. First it was assumed that rotation is around a point r align with the body and calculate the variables that fulfil conditions mentioned above. The body length is equal to L.

$$F_x + F_{meso} + F_{meta} = 0 \tag{C.1}$$

$$F_x(r+L) + F_{meso}(r + L\eta_{meso}) + F_{meta} r = 0 \tag{C.2}$$

In matrix notation,

$$\begin{bmatrix} 1 & 1 \\ r + L\,\eta_{meso} & r \end{bmatrix} \begin{bmatrix} -F_x \\ -F_x(r+L) \end{bmatrix} = \begin{bmatrix} F_{meso} \\ F_{meta} \end{bmatrix} \tag{C.3}$$

The above equation is useful because the determinant of the matrix is equal to $L\eta_{meso}$, thus these equations do not hold for $\eta_{meso} = 0$. The resulting displacement at each segment is calculated with Hooke's law, $F = -kx$.

$$x_{meso} = \frac{F_x}{\eta_{meso} k_{meso}} \tag{C.4}$$

$$x_{meta} = \frac{(\eta_{meso} - 1) F_x}{\eta_{meso} k_{meta}} \tag{C.5}$$

From these equations, it is possible to calculate where the position of points along the body. The body is represented by a line equation,

$$B_y = \frac{(B_x - x_{meta}) \eta_{meso} L}{(x_{meso} - x_{meta})} \tag{C.6}$$

Note that the metathorax was initially at position $B = [0, 0]$ and that moves little along $\hat{y}$. Thus, to find η_{AOR} it is just necessary to find the new $B_y(B_x = 0)$. By substituting C.4 and C.5 in C.6 and dividing by L to get the relative position,

$$k_{meta} = w k_{meso} \tag{C.7}$$

$$\eta_{AOR} = \left[\frac{(\eta_{meso} - 1) \eta_{meso}}{\eta_{meso} - 1 - w} \right] \tag{C.8}$$

C.2 Subordination Rear Calibration

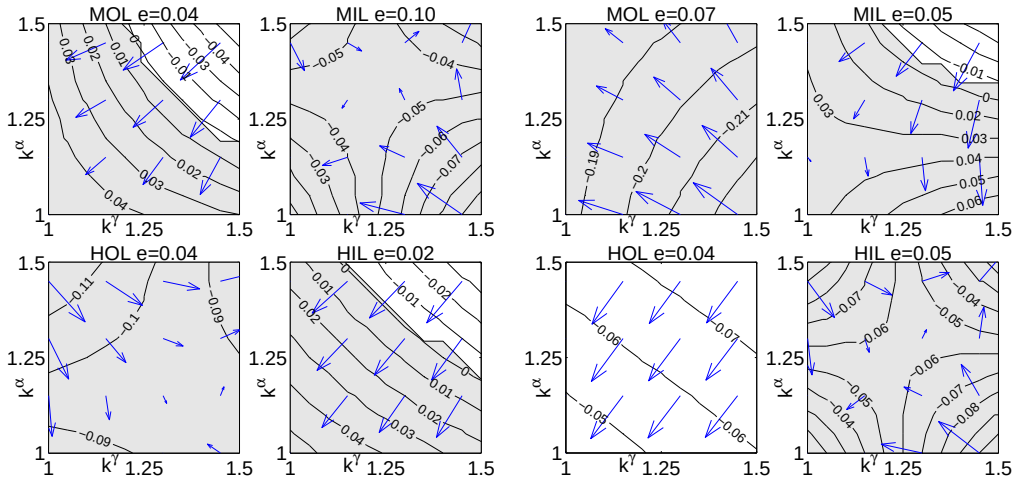
Leg swing statistics provide information on what are leg contributions to turn. The study concentrated on four parameters, angular acceleration at PEP, and AEP, average angular speed and average angular acceleration during swing. This parameters for the four middle and hind legs, i.e., middle and hind legs. Each of the variables, active k^{α}, k^{γ} and passive s^{α}, s^{γ}, was given three different values. All the results for variation of the active parameters are shown in table C.1.

However, it is difficult to discern what the results are, therefore, to aid visualisation, the pattern of results were fit to the equation C.9,

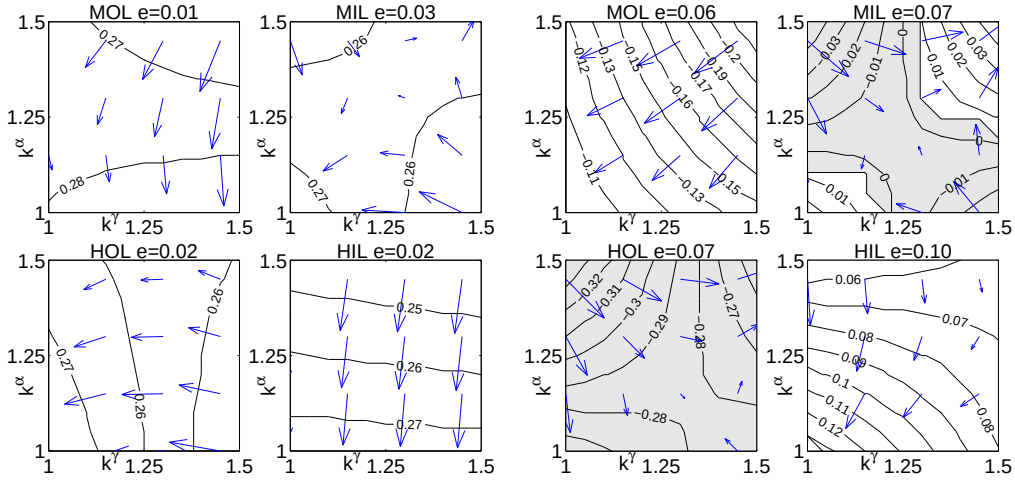
$$f(x_1, x_2) = a + b x_1 + c x_2 + d x_1 x_2 \tag{C.9}$$

Where x_i represented one of the parameters tested[1]. Fitted functions for table C.1 are shown in Figure C.1.

[1] No more terms were added to equation C.9 to over fitting.



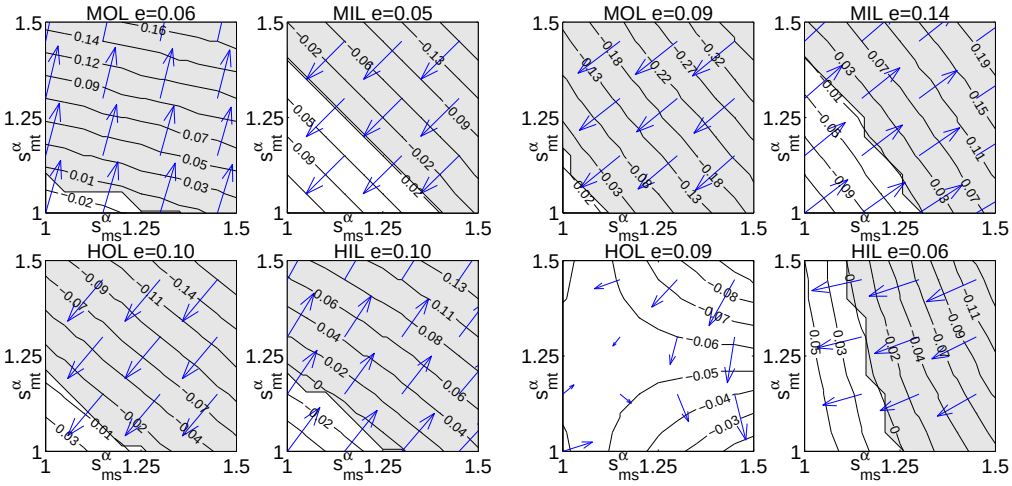
(a) PEP (b) AEP



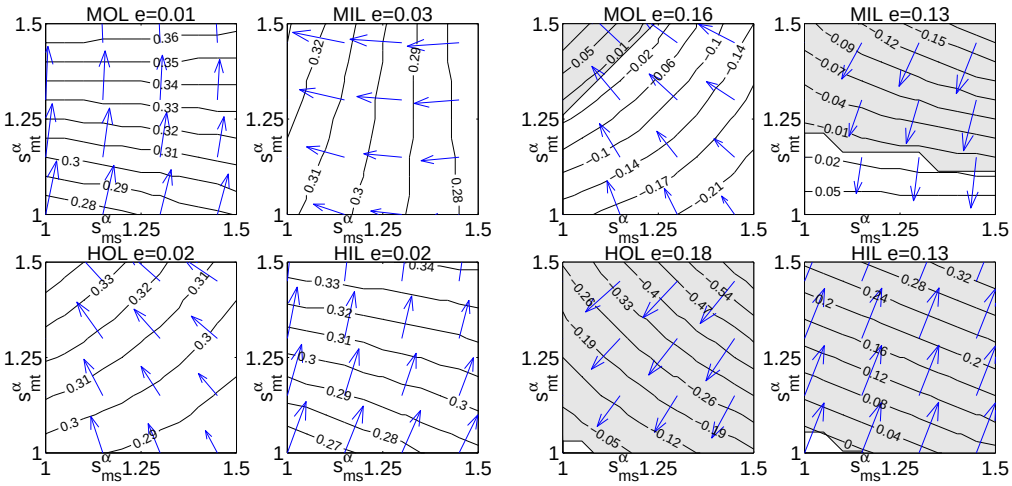
(c) $\dot{\theta}_B$ (d) $\ddot{\theta}_B$

Figure C.1: Active parameters calibration: κ^{α} and κ^{γ}

Each subfigure shows how the acceleration or speed due to each leg changes when varying k^{γ} (x-axis) and k^{α} (y-axis), with the results shown as a plane fitted to the data according to equation C.9. The error of this fitting is shown to the right of the title. Within each subfigure, the top graphs are middle legs and lower graphs are hind legs, with the inner legs on the right and outer legs in the left. The shaded area represents results corresponding to those observed in the insect.



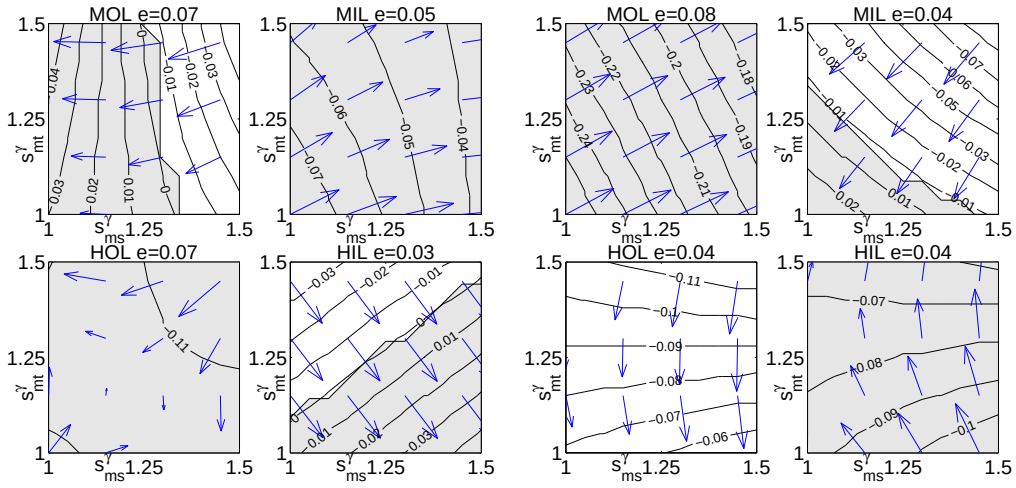
(a) PEP (b) AEP



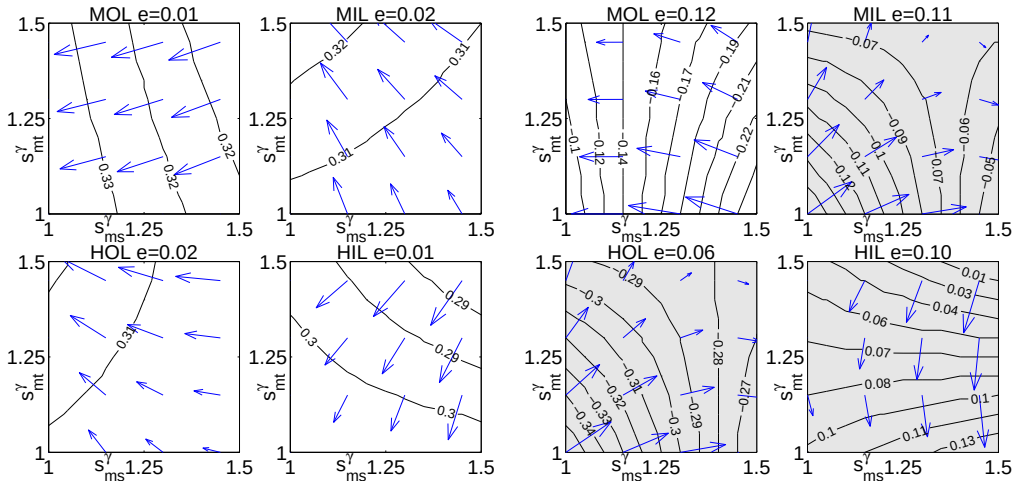
(c) $\dot{\theta}_B$ (d) $\ddot{\theta}_B$

Figure C.2: Subordination calibration: s^{α}_{meso} and s^{α}_{meta}

Each subfigure shows how the acceleration or speed change when varying s^{α}_{meso} (x-axis) and s^{α}_{meta} (y-axis) with the results shown as a plane fitted to the data according to equation C.9. The error of this fitting is shown to the right of the title. Within each subfigure, the top graphs are middle legs and lower graphs are hind legs, with the inner legs on the right and outer legs in the left. The shaded area represents results corresponding to those observed in the insect.



(a) PEP (b) AEP



(c) $\dot{\theta}_B$ (d) $\ddot{\theta}_B$

Figure C.3: Subordination calibration: s^{γ}_{meso} and s^{γ}_{meta}

Each subfigure shows how the acceleration or speed change when varying s^{γ}_{meso} (x-axis) and s^{γ}_{meta} (y-axis) with the results shown as a plane fitted to the data according to equation C.9. The error of this fitting is shown to the right of the title. Within each subfigure, the top graphs are middle legs and lower graphs are hind legs, with the inner legs on the right and outer legs in the left. The shaded area represents results corresponding to those observed in the insect.

			k^{γ}					
			Outer Leg			Inner Leg		
			1.0	1.3	1.5	1.0	1.3	1.5
PEP, k^{α}	Middle	1.5	0.03	-0.02	-0.05	-0.10	-0.11	0.00
		1.3	0.04	0.01	-0.05	-0.03	-0.07	-0.02
		1.0	0.05	0.06	0.04	-0.04	0.00	-0.10
	Hind	1.5	-0.09	-0.12	-0.08	0.02	0.00	-0.05
		1.3	-0.09	-0.12	-0.08	0.03	0.01	-0.02
		1.0	-0.09	-0.11	-0.13	0.06	0.02	0.02
AEP, k^{α}	Middle	1.5	-0.23	-0.24	-0.20	0.08	0.03	-0.06
		1.3	-0.18	-0.19	-0.14	0.03	0.03	-0.03
		1.0	-0.19	-0.19	-0.19	0.04	0.04	0.02
	Hind	1.5	-0.05	-0.05	-0.08	-0.11	-0.08	0.00
		1.3	-0.07	-0.07	-0.08	-0.04	-0.07	-0.03
		1.0	-0.04	-0.03	-0.07	-0.01	-0.08	-0.09
$\dot{\theta}_B$, k^{α}	Middle	1.5	0.29	0.28	0.26	0.24	0.26	0.27
		1.3	0.29	0.28	0.27	0.28	0.25	0.26
		1.0	0.28	0.29	0.27	0.27	0.26	0.26
	Hind	1.5	0.25	0.26	0.26	0.27	0.26	0.24
		1.3	0.27	0.25	0.26	0.28	0.25	0.24
		1.0	0.27	0.26	0.27	0.27	0.27	0.24
$\ddot{\theta}_B$, k^{α}	Middle	1.5	-0.15	-0.22	-0.22	-0.03	-0.02	0.05
		1.3	-0.11	-0.11	-0.19	0.01	0.00	0.05
		1.0	-0.11	-0.10	-0.13	0.01	-0.01	-0.08
	Hind	1.5	-0.26	-0.32	-0.22	0.07	0.07	0.11
		1.3	-0.30	-0.29	-0.30	0.15	0.02	0.02
		1.0	-0.26	-0.31	-0.34	0.13	0.09	0.08

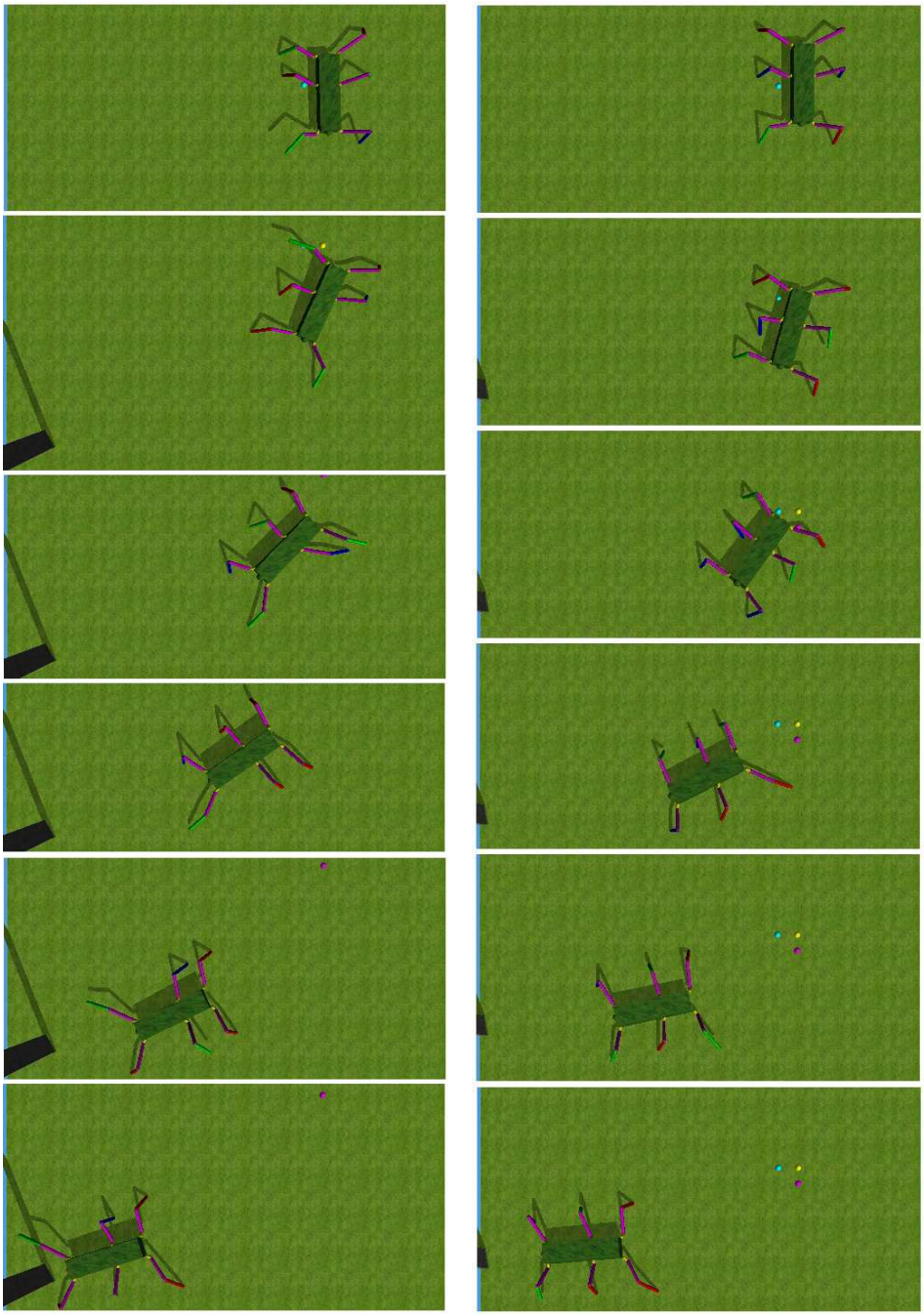
Table C.1: Nine acceleration results at the PEP of each leg when varying k^{α} and k^{γ}. Also the average speed $\dot{\theta}_B$ and the average acceleration $\ddot{\theta}_B$ while that legs is swinging.

Appendix D

Simulation Sequences

Figure D.1: Robot simulation snapshots. 180 degrees tight turn

The ‘visual’ target in this demonstration is at half the usual distance, approximately at a body length distance. The manoeuvre, if necessary, would not be initiated by vision. However, the important result is that rotation on the spot is feasible without variations in the basic controller.



(a) 60 (b) 90

Figure D.2: Robot simulation snapshots. Turning at 60 and 90 degrees

Two different simulations divided in six representative frames. Left: the simulation is controlled only by front legs and the middle and hind legs follow passively; target is at 60 degrees. Right: Complete thoracic differentiation implemented. Note that for this sequence the region shown is the same throughout, thus, it is easier to appreciate how the prothorax moves almost horizontal.

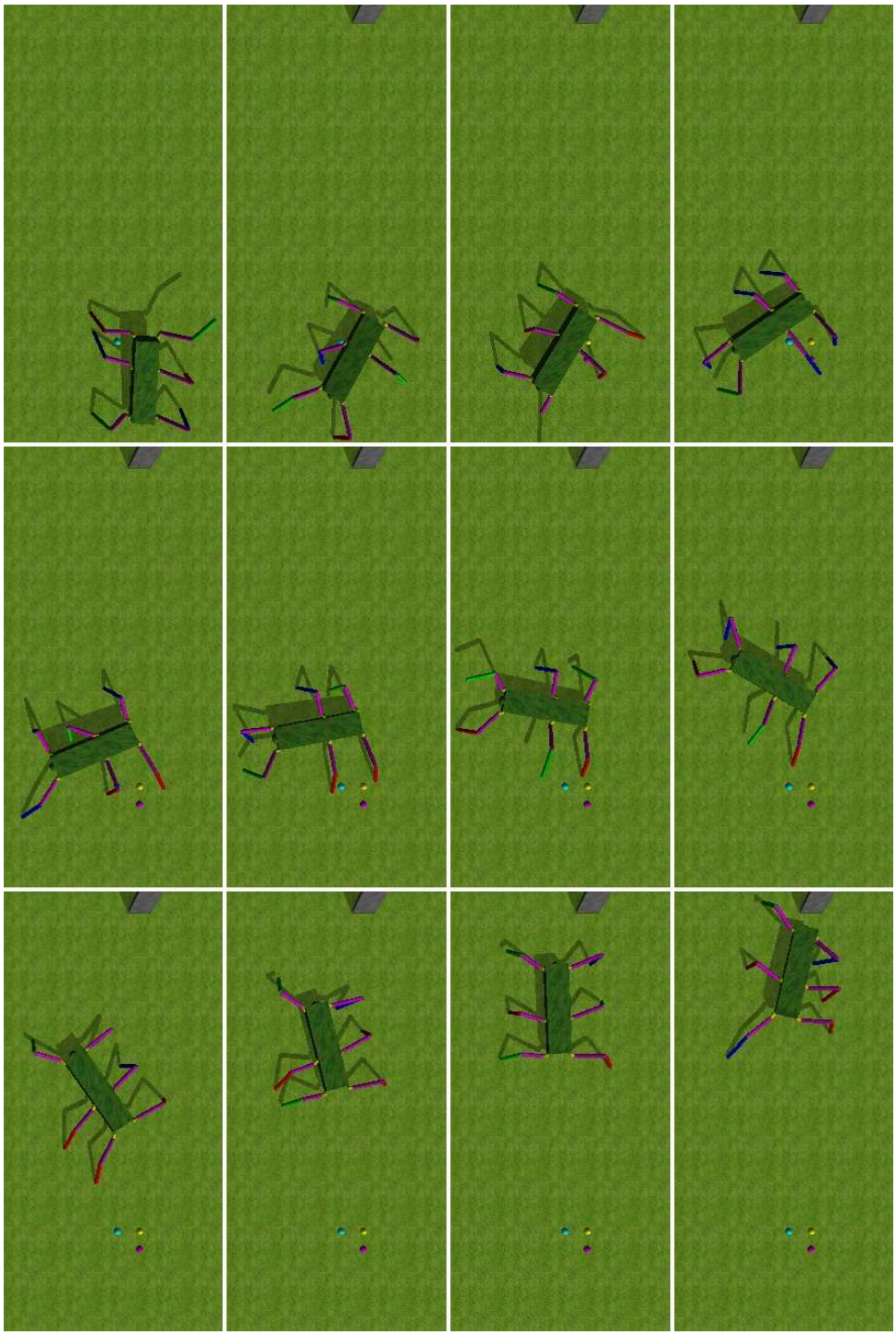


Figure D.3: Robot simulation snapshots. Turning at 180 degress

Frames run from left to right and from top to bottom. The target appears at 180 degrees, thus the direction (left or right) of the turn is random. During turn there was no changes in strategies implemented in any leg.

Appendix E

Replacing Jacobian

E.1 ANN Training Pseudocode

ANNs in chapter 6 were trained with algorithm E.1.1. The parameters were the function, the gradient of the function and initial values for the network weights. The algo-

rithm shown below finds the values of weights that minimize the function given.

Algorithm E.1.1: MINIMIZER($Function, Gradient, iniW$)

$W \leftarrow iniW$
$counter \leftarrow 0$
$speed \leftarrow iniSpeed$
while $(counter < MAX)$ **and** $(realValue > MIN)$
do
- $G_{prev} = G$
- $G \leftarrow Gradient(Ws)$
- optional gradient version
  - $G \leftarrow G + G_{prev}(G \cdot (G - G_{prev})/(G_{prev} \cdot G_{prev}))$
  - $G \leftarrow G + G_{prev}(G \cdot G)/(G_{prev} \cdot G_{prev}))$
- $getOut \leftarrow 0$
- **while** $true$
  - **do**
    - $W \leftarrow W - speed\,G$
    - $realValue_{prev} \leftarrow realValue$
    - $realValue \leftarrow Function(Ws)$
    - $error \leftarrow realValue - realValue_{prev}$
    - **if** $getOut = 1$
      - **then break**
      - **else**
        - **if** $error > 0$
          - **then**
            - $speed \leftarrow -speed$
            - **if** $speed = iniSpeed$
              - **then** $G \leftarrow Gradient(Ws)$
            - $getOut = 1$
          - **else** $speed \leftarrow 1.05\,speed$

The function and the gradient of the function given to algorithm E.1.1 were in fact pointers to variable functions. These were squared errors of the difference between the training data set and the values calculated by the different networks described chapter 6. The two possible activation functions were the Gaussian $G(x)$, or the sigmoid function $s(x)$, these are shown below.

$$G(x) = ae^{\frac{-(x-b)^2}{2c^2}} \tag{E.1}$$

$$S(x) = \frac{1}{1+e^{-x}} \tag{E.2}$$

E.2 Optimal AEP

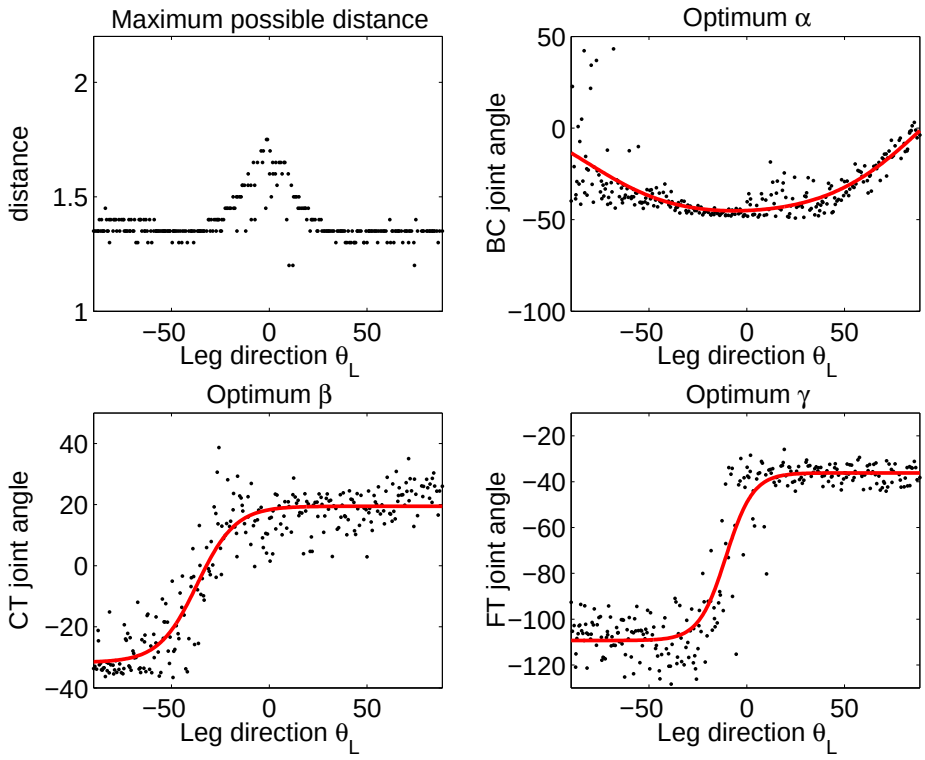


Figure E.1: Mesothorax optimal AEP angles fitting

Positive leg direction angles pull the leg towards the body; negative values away. Angles are shown in degrees. Top left: At a given leg direction range $\theta_{\bar{L}} + \Delta$ the maximum distance the leg could travel was found. Top right: Optimal angle for the α joint, this function is the most complex of all joints. Bottom Right: Optimal value for the γ joint, it basically only takes two extreme values. Bottom Left: Optimal value for the β joint. Although this joint has an optimal value, it is used to control local height.

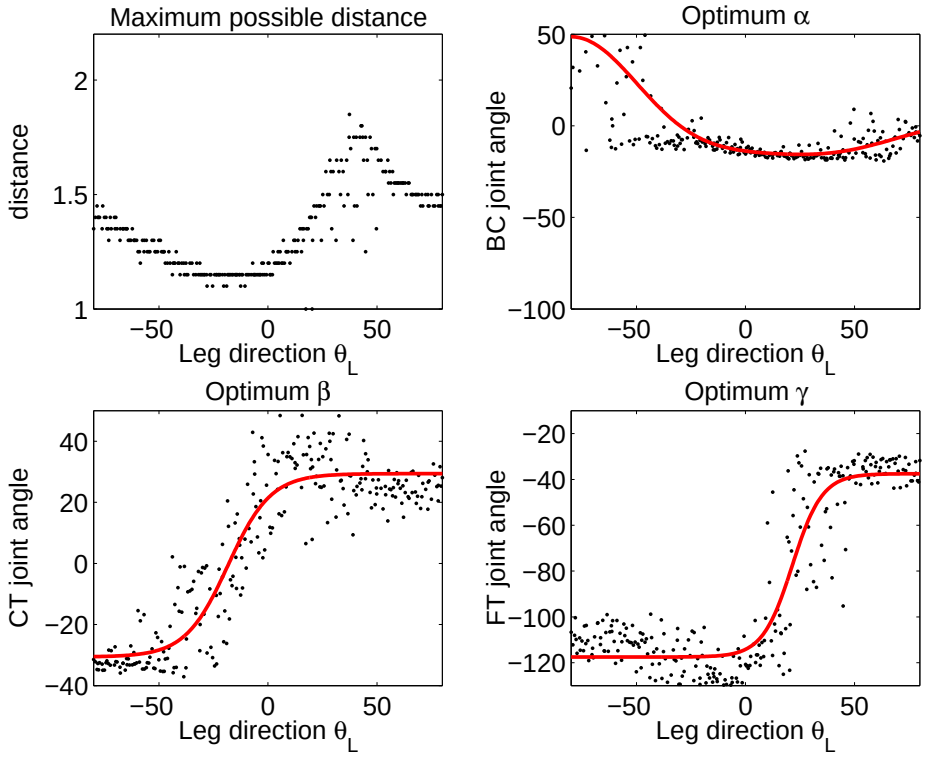


Figure E.2: Metathorax optimal AEP angles fitting

Positive leg direction angles pull the leg towards the body; negative values away. Angles are shown in degrees. Top left: At a given leg direction range $\theta_{\vec{L}} + \Delta$ the maximum distance the leg could travel was found. Top right: Optimal angle for the α joint, this function is the most complex of all joints. Bottom Right: Optimal value for the γ joint, it basically only takes two extreme values. Bottom Left: Optimal value for the β joint. Although this joint has an optimal value, it is used to control local height.

E.3 Validating Optimal Angle Joints

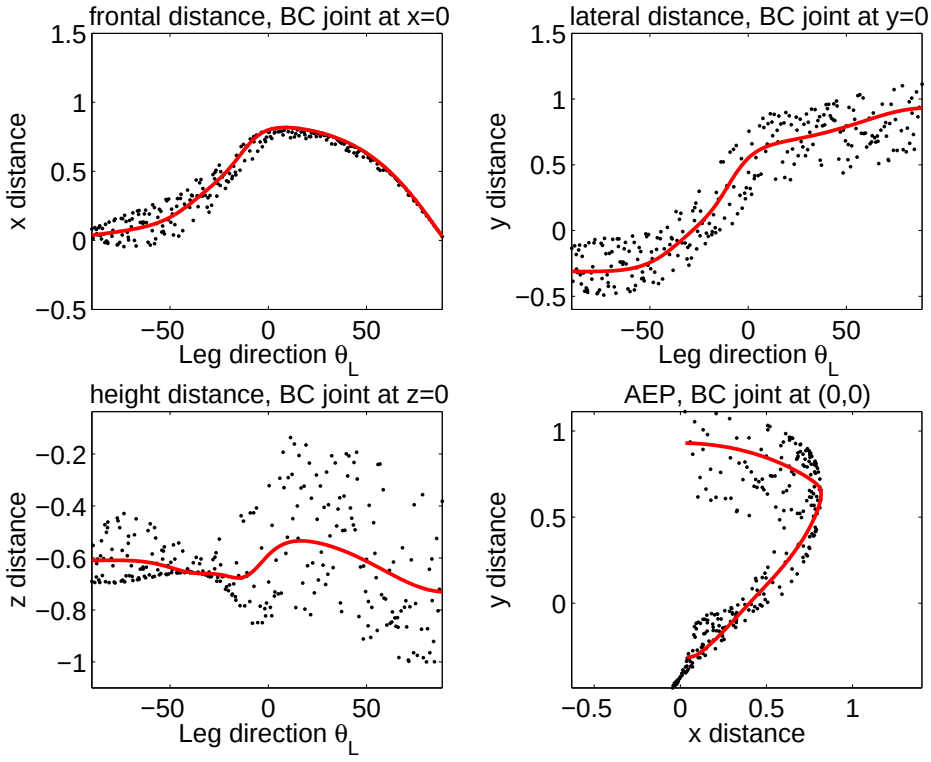


Figure E.3: Mesothorax optimal AEP with variations in the CT joint

Positive leg direction angles pull the leg towards the body; negative values away. Angles are shown in degrees. The coxa is at position $[0, 0, 0]$. Optimal values are shown by a solid line, dots represent variation due to β. Top left: positive values of x points to the front. Top right: positive values of y points away from the body. Bottom right: top view of the AEP. Note that β does not have a strong effect. Bottom left: Variation in the height of the AEP when β is moved by ± 20.

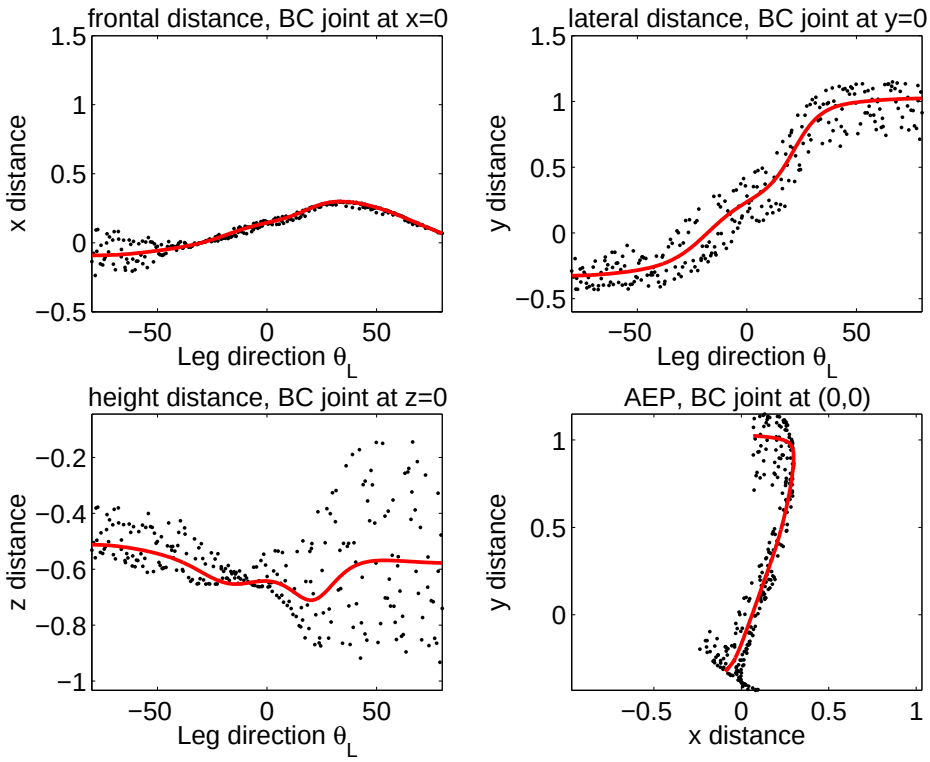


Figure E.4: Metathorax optimal AEP with variations in the CT joint

Positive leg direction angles pull the leg towards the body; negative values away. Angles are shown in degrees. The coxa is at position $[0, 0, 0]$. Optimal values are shown by a solid line, dots represent variation due to β. Top left: positive values of x points to the front. Top right: positive values of y points away from the body. Bottom right: top view of the AEP. Note that β does not have a strong effect. Bottom left: Variation in the height of the AEP when β is moved by ± 20.

List of Symbols

Variable Notation

g	Scalar
g_t	Variable at time t
$\vec{g}$	Vector
Δg	$\Delta g = g_t - g_{t-1}$
dg	Derivative of g
$\dot{g}, \ddot{g}$	First dg/dt and Second d^2g/dt^2 derivative with respect to time
$\hat{g}$	Unitary vector, normally a base vector
$\theta_{\vec{g}}$	Direction of vector $\vec{g}$ in polar coordinates
$r_{\vec{g}}$	Magnitude of vector $\vec{g}$
$\vec{g}_x$	x component of variable g, $g_x = g \cdot \hat{x}$
g_{pro}, g_{meso}, g_{meta}	Variable at the pro-, meso- and meta- thoracic segment
g^{α}, g^{β}, g^{γ}	Variable at the BC, CT and FT joint

Body position

$Length$	Body length
ψ''	Rotation of coxa segment around its elongated axis
ψ'	Rotation of coxa segment around body axis
α	BC joint angular position
β	CT joint angular position
γ	FT joint angular position
η	Relative position along the body, $\eta_{meta}(\eta = 0)$ at the metathorax and $\eta_{pro}(\eta = 1)$ at the prothorax

$\vec{B}_\eta$	Velocity at relative position along the body

Leg control variables

$\vec{T} = [T_x, T_y, T_z]$	Tarsus position in rectangular coordinates
$\vec{W} = [\alpha, \beta, \gamma]$	Leg joint angular positions
$r_{\vec{L}}$	Tarsus speed
$\theta_{\vec{L}}$	Tarsus angle direction relative to the body
$dz_{\vec{L}}$	Tarsus vertical direction
$h_{\vec{L}}$	BC local height
$\vec{L}(r_{\vec{L}}, \theta_{\vec{L}}, dz_{\vec{L}})$	Tarsus velocity in cylindrical base coordinates
$\vec{W}_d = [\alpha_d, \beta_d, \gamma_d]$	Leg deviation
$\vec{W}_{sp}$	Leg set-point
$J(\vec{T})$	Jacobian of the tarsus position

Visual Target Analysis

θ_T	Absolute angle position of the visual target
θ_P	Absolute prothorax angular position
θ_M	Absolute Metathorax angular position
θ_B	Absolute body angular heading
ϕ	Relative angle to the visual target
$\vec{v}_P = [\theta_{\vec{v}_P}, r_{\vec{v}_P}]$	Prothorax velocity relative to the body
$\vec{v}_M = [\theta_{\vec{v}_M}, r_{\vec{v}_M}]$	Metathorax velocity relative to the body

Stepping Parameters

f_S	Step frequency
T_S	Step Period
T_r	Time of leg retraction
T_p	Time of leg protraction
v_L	Left leg speed
v_R	Right leg speed
f_{SL}	Step frequency on the left
f_{SR}	Step frequency on the right

Stance control

$\theta_{sp,t}$	Joint angular set-point at time t
$\theta_{d,t}$	Angular debiation a time t
κ	Active control modulating angular joint activation
Φ	Activation of the joint, e.g., angle to the target
s	Subordination: parameter modulating angular debiation at the joint
$\vec{S} = [s^{\alpha}, s^{\beta}, s^{\gamma}]$	Vector containing the subordination values of the three joints

Tracking Algorithm Parameters

$\vec{M}$	Mark position. Normally tarsus on stance
k	Sequence frame label
K	Total frames in the sequence
N	Total marks at frame k
$\vec{\Upsilon}_k$	Translation of marks at frame k
$R_k(\theta_k)$	Rotation of marks at frame k by angle θ_k
$H(\vec{\Upsilon}, \theta)_k$	Homogeneous transformation at frame k translation then rotation
σ_k	Scale applied to marks at frame k
$\vec{M}_{k,i}$	Mark i at frame k, i=$\{0..N\}$
$\vec{M}^*_k$	Average mark position at frame k

Glossary / Index

Central Pattern Generator (CPG) Neuronal network capable of generating a rhythmic pattern without phasic input, 9, 14, 19, 23, 29, 33, 35, 39, 42

Centre of mass (COM) In a uniform gravitational field the COM (or centroid) is a point of balance, 22, 23, 32, 45, 50, 53, 55, 74, 103, 119, 120, 122, 148, 164, 170, 173, 189

Chordotonal organ (CO) stretch receptors in insects. They are used to detect the position of the body segments and appendages, or, in tympanal organs, the vibrations caused by sounds, 11, 17, 18

Contralateral On the opposite side of the body, 21, 23, 35, 45, 67, 71, 120, 125, 164

Convex-hull Minimal convex set containing all points in a set, 23, 53

Coxa The first segment of the leg, 15, 16, 19, 31, 70, 99, 121

Coxa-trochanteral joint (CT) Articulation joining the coxa and the trochanter. The latter is fused with the femur in the stick insect, 15–17, 19, 30, 37, 39, 81, 82, 84, 102, 152, 168

Curvature The ratio of the change in the angle of a tangent that moves over a given arc to the length of the arc, 38, 45, 46, 74, 132

Depressor Muscle group moving the femur down around the CT joint, 17, 30, 82

Differential drive Wheel axis with an arrangement of gears in an epicyclic train permitting the rotation of two shafts at different speeds, 45

Dynamic Branch of classical mechanics that is concerned with the effects of forces on the motion of objects, i–iii, 5, 6, 25, 35, 39, 42, 45, 70, 79, 98, 115, 137, 151, 152, 154, 155, 159, 163

Excitatory postsynaptic potential Increase in postsynaptic membrane potential caused by the flow of positively charged ions into the postsynaptic cell, 29, 30, 119, 121

Exoskeleton A hard outer structure that provides protection or support for an organism, 17–19, 31, 84

Extensor Muscle group moving the tibia away from the body around the FT joint, 17, 29, 30

Exteroceptors Sensory neurons outside the exoskeleton, 18

Femoral chordotonal organ Chordotonal organ monitoring the femur-tibia joint, 18, 19, 28, 30

Femur The second segment of the leg, 15, 18, 19, 30, 99

Femur-tibia joint (FT) Articulation joining the femur and the tibia, (knee), 15–19, 29, 30, 37, 39, 73, 75, 81, 82, 84, 102, 107, 111, 121, 134, 136, 153, 160, 167–169

Flexor Muscle group moving the tibia towards the body around the FT joint, 17, 30, 82, 169

Force controller Controller correcting for error in the torque relative to a force set point, 82

Front inner leg (FIL) Front leg on the inner side of the turn, 58, 169

Front legs Anterior pair of legs positioned at the prothorax, i, 10, 22–26, 43, 44, 55, 58, 61, 62, 66, 70, 73–76, 79, 103, 108, 110, 117, 120, 121, 124, 128, 130, 131, 136, 137, 139–141, 146, 164, 165, 167, 170, 189

GA Genetic Algorithm, 36, 42, 123, 125, 168

Gait Walking stable pattern, 23, 27, 32, 34–36, 38, 53, 66, 78, 118, 124, 125, 128, 164, 168

Ganglia A group of nerve cells forming a nerve centre, 7, 28, 29, 31, 35, 39, 41, 45, 76, 77, 79, 85, 150, 162–164, 167, 169

Hair plait Clusters of mechanosensitive hairs, usually positioned close to joints, which function as proprioceptors for joint movement, 18

Hind inner leg (HIL) Hind leg on the inner side of the turn, 26, 64, 66, 71, 73, 124, 137, 139, 143, 147, 168

Hind legs Posterior pair of legs positioned at the metathorax, i, 22–25, 44, 58, 64, 70, 71, 73, 100, 108, 110, 117, 120, 121, 124, 128, 136, 137, 139, 141, 165, 170, 192

Hind outer leg (HOL) Hind leg on the outer side of the turn, 66, 71, 73, 137, 139, 140

Inhibitory postsynaptic potential A postsynaptic potential is considered inhibitory when the resulting change in membrane voltage makes it more difficult for the cell to fire an action potential, 29, 30, 119, 120, 125, 164

Instant Axis of Rotation (IAR) In a body which has motions both of translation and rotation, is a line, which is supposed to be rigidly united with the body, and which for the instant is at rest, 46, 48–50, 53, 55, 58, 64, 67–69, 71, 74, 130, 131, 164, 174

Interneurons Nerve cells that are not sensory or motor neurons and are subvided in *relay* or *projection interneurons*, 7, 13, 28–30, 35, 39, 79, 155–157, 159, 160

Interoceptors Sensory neurons inside the exoskeleton, 18

Ipsilateral On the same side on the body, 21–23, 35, 119, 124

Jacobian matrix The Jacobian matrix is the matrix of all first-order partial derivatives of a vector-valued function. It represents the best linear approximation to a differentiable function near a given point, 85, 86, 156, 160, 164, 166, 187

Junction potential Potential difference at the synapse. It can be excitatory or inhibitory, 29, 30

Kinematic The branch of mechanics dealing with the study of the motion of a body or a system of bodies without consideration given to its mass or the forces acting on it, i, 10, 13, 26, 27, 34, 35, 38, 40, 42, 45, 53, 68, 70, 71, 73–75, 78, 79, 151, 159, 161, 164, 166

Levator Muscle group moving the femur up around the CT joint, 17, 30

Linear A function f(x) is linear if satisfies the properties of additivity f(x+y)=f(x)+f(y), and homogeneity f(ax)=af(x), 70, 81, 94, 108, 155, 156, 159, 160, 174

Mechanosensor Sensory neuron activated in response to mechanical pressures or distortions, 28, 39

Mesothorax The middle of the three divisions of the thorax of an insect, 15, 19, 58, 61, 68, 71, 73–75, 88, 99, 117, 131–134, 136, 137, 139–142, 146, 147, 152–154, 165, 167, 170, 191

Metathorax The hindmost of the three divisions of the thorax of an insect, 15, 19, 53, 55, 58, 61, 62, 68, 69, 71, 73–75, 88, 99, 117, 120, 128, 130–133, 136, 137, 139–142, 146–148, 152–154, 160, 165, 167, 191, 192

Middle inner leg (MIL) Middle leg on the inner side of the turn, 44, 66, 71, 73, 137, 139, 147, 169

Middle legs Pair of legs positioned at the mesothorax, i, 22–25, 29, 39, 44, 64, 70, 71, 73, 105, 110, 117, 128, 136, 165, 170, 192

Middle outer leg (MOL) Middle leg on the outer side of the turn, 66, 71, 73, 137, 139

Morphology Outward appearance of an organism or taxon and its component, 4, 6, 7, 13–15, 29, 34, 37, 100, 122, 148, 162, 189

Motoneurons Nerve cells carrying commands to muscles and glands, 7, 28–30, 35, 39, 86, 150, 164, 169

Multi-layer perceptron This class of networks consists of multiple layers of computational units, 155

Multipolar A multipolar neuron is a type of neuron that possesses a single (usually long) axon and many dendrites, allowing for the integration of a great deal of information from other neurons, 18

Negative feedback The feedback signal, usually the error, is subtracted to the current input, i, 20, 37, 40, 41, 74, 75, 77–79, 82, 83, 85, 86, 97, 110, 111, 113, 163, 164

Nervous system The system of cells, tissues, and organs that regulates the body's responses to internal and external stimuli, 6, 7, 13, 28, 31, 39, 41, 162

Neural Network A real or virtual device, modelled after the human brain, in which several interconnected elements process information simultaneously, adapting and learning from past patterns, 6, 11, 20, 30–32, 34, 35, 38, 150, 151

Neuroethology Is a branch of neuroscience that emphasizes the study of neural mechanisms of natural behavior, 7

Neuron, nerve cell Electrically excitable cells in the nervous system that process and transmit information, 7, 20, 21, 28–32, 39, 41, 123, 150, 155, 161, 166

Neurophysiology Neurophysiology is the study of functioning of the nervous system, i, 8, 9, 14, 30–32, 39, 40, 74, 169

Neuropile Fibrous network of unmyelinated nerve fibers, 28

Non-linear Nonlinear systems behaviour is not expressible as a sum of the behaviours of its descriptors, 16, 71, 74, 85, 90, 108, 154–156

Non-spiking neuron Neuron incapable of producing an action potential, 28–30, 35, 39, 150

Nonholonomic system Motion is a n-dimensional space by controlling n variables. Path dependent, 3, 4

NSI Non-spiking interneuron, 29, 30

Open loop Type of controller which computes its input into a system using only the current state and its model of the system, 85, 87, 169

Perceptron The perceptron is a kind of binary classifier. It can be seen as the simplest kind of feedforward neural network, 155

Phasic response Information processing that depends on variations in input value, 18

Position controller Controller correcting for error in the position relative to a position set point, 82, 85, 87, 92, 94, 100, 101, 113–115

Positive feedback The feedback signal is positively increased to the current input, i, 37, 40, 41, 74, 75, 78, 79, 83, 84, 86, 97, 110, 112, 113, 130, 163, 164, 168

Positive feedback The feedback signal, usually the error, is positively increased to the current input, 41, 82, 83

Posterior Extreme Position (PEP) Position where the leg changes from stance to swing, 6, 18, 27, 64, 66, 78–80, 108, 132, 137, 139, 152, 192

Propioceptive Sensory neurons monitoring internal states, 30

Propioceptor A sensory receptor that detects the motion or position of the body or a limb by responding to stimuli arising within the organism, 18

Proportional derivative (PD) Controller correcting the error proportionally damped by a derivator factor, 82, 95, 187

Prothorax The anterior division of the thorax of an insect, 15, 19, 53, 55, 58, 61, 62, 66, 68–71, 73–76, 79, 99, 128, 131–133, 140–142, 146, 148, 152, 153, 155, 160, 161, 165, 191

Protractor Muscle group moving the coxa anteriorly around the BC joint, 17

Radial Basis Is a function which has built into it a distance criterion with respect to a centre, 155

Relaxation oscillator Free running circuit that outputs pulses with a period dependent on internal states, 20, 37

Retractor Muscle group moving the coxa posteriorly around the BC joint, 17, 66

Scoloparium Chordotonal sensory unit, 18

Sensillum A simple sensory receptor consisting of one cell or a few cells, especially a hairlike epithelial cell projecting through the cuticle of arthropods, 18

Sensory neurons Nerve cells carrying information from the body's periphery into the nervous system, 6, 7, 9, 13, 14, 18–21, 28–31, 34, 39, 102, 106, 114, 120, 150

Set-point Is the set desired positioned that the controllers tries to reach and maintain, 40, 85, 86, 110, 112, 187

Sigmoid function Refers to the special case of the logistic function that produces a sigmoid curve, a curve having an "S" shape, 153, 155

Spiking Neuron Neuron capable of producing an action potential (see Action potential), 28, 30, 39, 123, 150, 166

Stability polygon The convexhull formed by legs in contact with the ground, 53

Stance phase Step phase when the leg is in contact with the ground. a.k.a, return stroke or power stroke, 6, 18, 20–23, 26, 27, 36, 37, 47, 50, 53, 58, 64, 66–68, 75–82, 97, 102, 103, 110–112, 118–122, 124, 125, 132, 143, 150–152, 161, 162, 164, 168, 173, 174, 182

Steady state error Marginal error when the time approaches infinity, 85, 92, 95

Step phase Stepping cycle is divided in two phases, stance and swing, 6, 20–22, 27, 35, 66, 67, 79, 81, 82, 112, 118–121, 123, 162, 168, 182

Step selector Algorithm or neural network controlling changes in step phase, 81, 118

Stick insect Phasmatodea. Usually categorised as a suborder of Orthoptera. Herein refers to the specie *Carausius morosus*, i–iii, 9–11, 13–16, 18–21, 23–27, 31, 32, 34–36, 38, 39, 41–44, 46, 54, 55, 57, 61, 68, 73–76, 79, 84, 97, 108, 113, 122, 138, 148, 150, 162, 164, 166, 169–171

Swing phase Step phase when protracts and is not mechanically coupled with the ground, 6, 18, 20–23, 27, 30, 34, 36, 37, 47, 50, 64, 66, 67, 71, 78, 81, 82, 102, 105, 109–111, 118–120, 122, 124, 125, 139, 140, 143, 151, 152, 162, 164, 168, 181, 192

Synapse The point at which two neurons communicate. The nerve cell transmitting is the presynaptic cell; the cell receiving is the postsynaptic, 29, 30

Tarsus Last segment in the leg composed of many individual compartments. Arthropod's foot, 14–16, 19, 22, 23, 25–27, 31, 41, 43–47, 50, 53, 55, 61, 62, 66–68, 70, 73–75, 78–82, 84, 85, 97, 103, 107, 108, 110, 117–122, 132, 136, 137, 139–141, 152–154, 160, 163, 164, 170, 171, 176, 189

Tetrapod gait Walking gait characterised by at least four legs on stance, 23, 27, 36, 66, 124, 125, 168

Tibia The third segment of the leg, 15, 19, 30, 99, 121

Tonic response Information processing that depends on absolute input value, 18

Tripod gait Walking gait characterised by at least three legs on stance, 23, 27, 36, 38, 125, 164, 168

Velocity controller Controller correcting for error in the velocity relative to a velocity set point, 82, 83, 85, 97, 110, 113, 163

WalkNet Kinematic stick insect based simulation implemented with ANN, 36–38, 42, 132

Bibliography

Akay, T., Bässler, U., Gerharz, P., and Büschges, A. (2001). The role of sensory signals from the insect coxa-trochanteral joint in controlling motor activity of the femur-tibia joint. *The Journal of Physiology*, 85:594–604.

Albiez, J., Luksch, T., Berns, K., and Dillmann, R. (2003). Reactive reflex-based control for a four-legged walking machine. *Robotics and Autonomous System*, 44:181–189.

Altendorfer, R., Moore, N., Komsuoglu, H., Buehler, M., Jr., H. B., McMordie, D., Saranli, U., Full, R., and Koditschek., D. (2001). Rhex: A biologically inspired hexapod runner. *Autonomous Robots*, 11:207–213.

Arena, P., Cruse, H., Fortuna, L., Frasca, M., and Patane, L. (2002). A cellular nonlinear approach to decentralized locomotion control of the stick insect. In *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, volume 4, pages IV–165 – IV–168.

Barfoot, T., Earon, E., and D'Eleuterio, G. (2000). A step in the right direction: Learning hexapod gaits through reinforcement. In *International Symposium on Robotics*, Montreal, Quebec.

Barnes, R., Calow, P., and Olive, P. (1993). *The Invertebrates, a new synthesis*. Blackwell Science.

Bartling, C. and Schmitz, J. (2000). Reaction to disturbances of a walking leg during stance. *J. Exp. Bio.*, 203:1211–1223.

Bässler, D., Büschges, A., Meditz, S., and Bässler, U. (1996). Correlation between muscle structure and filter characteristics of the muscle-joint system in three orthopteran insect species. *The Journal of Experimental Biology*, 199:2169–2183.

Bässler, U. (1976). Reversal of a reflex to a single motoneuron in the stick insect *Carausius morosus*. *Biological Cybernetics*, 24:47–49.

Bässler, U. (1977). Sense organs in the femur of the stick insect and their relevance to the control of position of the femur-tibia-joint. *The Journal of Comparative Physiology*, 121:99–113.

Bässler, U. (1988). Functional principles of pattern generation for walking movements of stick insect forelegs: The role of the femoral chordotonal organ afferents. *The Journal of Experimental Biology*, 136:125–147.

Bässler, U. and Büschges, A. (1998). Pattern generation for stick insect walking movements-multisensory control of a locomotor program. *Brain Research reviews*, 27:65–68.

Bässler, U., Foth, E., and Breutel, G. (1985). The inherent walking direction differs for the prothoracic and metathoracic legs of stick insects. *The Journal of Experimental Biology*, 116:301–311.

Bässler, U. and Stein, W. (1996). Contributions of structure and innervation pattern of the stick insect extensor tibiae muscle to the filter characteristics of the muscle-joint system. *The Journal of Experimental Biology*, 199:2185–2198.

Beer, R. D., Chiel, H. J., Quinn, R. D., Espenschied, K. S., and Larsson, P. (1992). A distributed neural network architecture for hexapod robot locomotion. *Neural Comput.*, 4(3):356–365.

Bennet-Clark, H. C. (1975). The energetics of the jump of the locust *SCHISTOCERCA GREGARIA*. *Journal of Experimental Biology*, 63:53–83.

Berns, K., Cordes, S., and Ilg, W. (1994). Adaptive, neural control architecture for the walking machine lauron. In *In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems.*, pages 1172–1177, Munich.

Berns, K., Dillmann, R., and Piekenbrock, S. (1995). Neural networks for the control of a six-legged walking machine. *Robotics and autonomous systems*, 14:233–244.

Bläsing, B. and Cruse, H. (2004a). Mechanisms of stick insect locomotion in a gap-crossing paradigm. *The Journal of Comparative Physiology A*, 190:173–183.

Bläsing, B. and Cruse, H. (2004b). Stick insect locomotion in a complex environment: Climbing over large gaps. *The Journal of Experimental Biology*, 207:1273–1286.

Brunn, D. E. (1998). Cooperative mechanisms between leg joints of *Carausius morosus* i. nonspiking interneurons that contribute to interjoint coordination. *Journal of Neurophysiology*, 79:2964–2976.

Brunn, D. E. and Dean, J. (1994). Intersegmental and local interneurons in the metathorax of the stick insect *Carausius morosus* that monitor middle leg position. *Journal of Neurophysiology*, 72(3):1208–1219.

Brunn, D. E. and Heuer, A. (1998). Cooperative mechanisms between leg joints of *Carausius morosus* ii. motor neuron activity and influence of conditional bursting interneuron. *Journal of Neurophysiology*, 79:2977–2985.

Bucher, D., Akay, T., DiCaprio, R. A., and Büschges, A. (2003). Interjoint coordination in the stick insect leg-control system: The role of positional signaling. *Journal of Neurophysiology*, 89:1245–1255.

Büsches, A., Schmitz, J., and Bässler, U. (1995). Rhythmic pattern in the thoracic nerve cord of the stick insect induced by pilocarpine. *The Journal of Experimental Biology*, 198:435–456.

Büschges, A. (1989). Processing of sensory input from the femoral chordotonal organ by spiking interneurons of stick insect. *The Journal of Experimental Biology*, 144:81–111.

Büschges, A. (1990). Nonspiking pathways in a joint-control loop of the stick insect *Carausius Morosus*. *The Journal of Experimental Biology*, 151:133–160.

Büschges, A. (1994). The physiology of sensory cells in the ventral scoloparium of the stick insect femoral chordotonal organ. *The Journal of Experimental Biology*, 189:285–292.

Büschges, A. (1998). Inhibitory synaptic drive patterns motoneuronal activity in rhythmic preparations of isolated thoracic ganglia in the stick insect. *Brain Research*, 783:262–271.

Büschges, A. (2004). Sensory control and organization of neural networks mediating coordination of multisegmental organs for locomotion. *Journal of Neurophysiology*, 93:1127–1135.

Calvitti, A. and Beer, R. D. (2000). Analysis of a distributed model of leg coordination i. individual coordination mechanisms. *Biological Cybernetics*, 82:197–206.

Camhi, J. M. and Johnson, E. N. (1999). High-frequency steering maneuvers mediated by tactile cues: antennal wall-following in the cockroach. *J. Exp. Bio.*, pages 631–643.

Clark, J. E., Cham, J. G., Bailey, S. A., Froehlich, E. M., Nahata, P. K., Full, R. J., and Cutkosky, M. R. (2001). Biomimetic design and fabrication of a hexapedal running robot. In *IEEE International Conference on Robotics and Automation*, pages 1–7.

Comer, C., Parks, L., and Halvorsen, M. (2003). The antennal system and cockroach evasive behavior. ii. stimulus identification and localization are separable antennal functions. *Journal of Comparative Physiology A*, 189:97–103.

Cruse, H. (1976a). The control of body position in the stick insect *(Carausius morosus)*, when walking over uneven surfaces. *Journal of Biological Cybernetics*, 24:25–33.

Cruse, H. (1976b). The function of the legs in the free walking stick insect, *Carausius Morosus*. *The Journal of Comparative Physiology*, 112:235–262.

Cruse, H. (1981). Is the position of the femur-tibia joint under feedback control in the walking stick insect? i. force measurements. *The Journal of Experimental Biology*, 92:87–95.

Cruse, H. (1985a). Which parameters control the leg movement of a walking insect? i. velocity control during the stance phase. *The Journal of Experimental Biology*, 116:343–355.

Cruse, H. (1985b). Which parameters control the leg movement of a walking insect? ii. the start of the swing phase. *The Journal of Experimental Biology*, 116:357–362.

Cruse, H. (1990). What mechanisms coordinate leg movement in walking arthropods. *Trends in Neurosciences*, 13:15–21.

Cruse, H. (2002). The functional sense of central oscillations in walking. *Integrative and Comparative Biology*, 86:271–280.

Cruse, H. and Bartling, C. (1995). Movement of joint angles in the legs of a walking insect, *Carausius morosus*. *Journal of Insect Physiology*, 41(9):761–771.

Cruse, H., Bartling, C., Cymbalyuk, J., Dean, J., and Dreifert, M. (1995). Modular artificial neural net for controlling a six-legged walking system. *Biological Cybernetics*, 72:421–430.

Cruse, H., Battling, C., Cymbalyuk, G., Dean, J., and Dreifert, M. (1994). A neural net controller for a six-legged walking system. In *From Perception to Action Conference, 1994., Proceedings*, pages 55–65.

Cruse, H., Dautenhahn, K., and Schreiner, H. (1992). Coactivation of leg reflexes in the stick insect. *Biological Cybernetics*, 67:369–375.

Cruse, H., Dean, J., Müller, U., and Schmitz, J. (1991). The stick insect as a walking robot. *Advanced Robotics Ŕobots in Unstructured Environments; 91 ICAR., Fifth International Conference on*, pages 936–940.

Cruse, H., Kindermann, T., Schumm, M., Dean, J., and Schmitz, J. (1998). Walknet-a biologically inspired network to control six-legged walking. *neural networks*, 11:1435–1447.

Cruse, H., Kühn, S., Park, S., and Schmitz, J. (2004). Adaptive control for insect leg position: controller properties depend on substrate compliance. *Journal of Comparative Physiology A*, 190:983–991.

Cruse, H. and Saavedra, M. G. S. (1996). Curve walking in crayfish. *J. Exp. Bio.*, pages 1477–1482.

Cruse, H., Schmitz, J., Braun, U., and Schweins, A. (1993). Control of body height in a stick insect walking on a treadwheel. *The Journal of Experimental Biology*, 181:141–155.

Cymbalyuk, G. S., Borisyuk, R. M., Mller-Wilm, U., and Cruse, H. (1998). Oscillatory network controlling six-legged locomotion: Optimization of model parameters. *neural networks*, 11:1449–1460.

Dean, J. (1989). Leg coordination in the stick insect *Carausius Morosus*: Effects of cutting thoracic connectives. *The Journal of Experimental Biology*, 145:103–131.

Dean, J. (1991). Effect of load on leg movement and step coordination of the stick insect *Carausius morosus*. *The Journal of Experimental Biology*, 159:449–471.

Delcomyn, F. (1999). Walking robots and the central and peripheral control of locomotion in insects. *Autnomous Robots*, 7:259–270.

Delcomyn, F. and Nelson, M. E. (2000). Architectures for a biomimetic hexapod robot. *Robotics and Autonomous Systems*, 30:5–15.

DiCaprio, R. A., Wolf, H., and Büschges, A. (2002). Activity-dependent sensitivity of propioceptive sensory neurons in the stick insect femoral chordotonal organ. *Journal of Neurophysiology*, 88:2387–2398.

Dickinson, M. H., Farley, C. T., Full, R. J., Koehl, M. A. R., Kram, R., and Lehman, S. (2000). How animals move: An integrative view. *SCIENCE*, 288:100–106.

Diederich, B., Schumm, M., and Cruse, H. (2002). Stick insects walking along inclined surfaces. *Journal of Biological Cybernetics*, 42:165–173.

Domenici, P., Jamon, M., and Clarac, F. (1998). Curve walking in freely moving crayfish (*PROCAMBARUS CLARKII*). *J. Exp. Bio.*, 201:1315–1329.

Dürr, V. (2001). Stereotypic leg searching movements in the stick insect: Kinematic analysis, behavioural context and simulation. *The Journal of Experimental Biology*, 204:1589–1604.

Dürr, V. (2005). Context-dependent changes in strength and efficacy of leg coordination mechanisms. *J. Exp. Bio.*, 208:2253–2267.

Dürr, V. and Ebeling, W. (2005). The behavioural transition from straight to curve walking: kinetics of leg movement parameters and the initiation of turning. *J. Exp. Bio.*, 208:2237–2252.

Dürr, V., König, Y., and Kittmann, R. (2001). the antennal motor system of the stick insect *Carausius morosus*: anatomy and antennal movement pattern during walking. *Journal of Comparative Physiology A*, 187:131–144.

Dürr, V., Krause, A. F., Schmitz, J., and Cruse, H. (2003). Neuroethological concepts and their transfer to walking machines. *The International Journal of Robotics Research*, 22(3–4):151–167.

Dürr, V. and Matheson, T. (2003). Graded limb targeting in an insect is caused by the shift of a single movement pattern. *J. Neurophysiol.*, 90:1754–1765.

Ehrlich, A. (1928). Vehicle propelled by steppers. US patent 1691233.

Espenschied, K. S., Quinn, R. D., Beer, R. D., and Chiel, H. J. (1996). Biologically based distributed control and local reflexes improve rough terrain locomotion in a hexapod robot. *robotics and autonomous systems*, 18:59–64.

Feynman, R. P. (1979). The Douglas Robb Memorial Lectures, Univeristy of Auckland.

Fielding, M. and Dunlop, R. (2004). Omnidirectional hexapod walking and efficient gaits using restrictedness. *The International Journal of Robotics Research*, 23:1105–1110.

Fielding, M., Dunlop, R., and Damaren, C. J. (2001). Hamlet: Force/position controlled hexapod walker - design and systems. In *Proceedings of the 2001 IEEE International*, volume 23, pages 984–989. Conference of Control Applications.

Fife, A., Laksanacharoen, S., Quinn, R., Beer, R., and Ritzmann, R. (2002). The evolution of neural network controllers for the targeted swing of cockroach-like robot. In *Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on*, volume 1, pages 929–933.

Filed, L. H. and Matheson, T. (1998). Chordotonal organs of insects. *Advances in Insect Physiology*, 27:1–228.

Fisher, H., Schmidt, J., Haas, R., and Büschges, A. (2001). Pattern generation for walking and searching movements of a stick insect leg. i. coordination of motor activity. *Journal of Neurophysiological*, 85:341–353.

Fortuna, L. and Patane, L. (2002). Hexapod locomotion control through a cnn based decentralized system. In *Industrial Electronics, 2002. ISIE 2002. Proceedings of the 2002 IEEE International Symposium on*, volume 4, pages 1312–1317.

Frantsevich, L. and Cruse, H. (1997). The stick insect, *Obrimus asperrimus* (phasmida, bacillidae) walking on different surfaces. *Journal of Insect Physiology*, 43(5):447–455.

Frantsevich, L. I. and Cruse, H. (2005). Leg coordination during turning on an extremely narrow substrate in a bug, *Mesocerus marginatus* (heteroptera, coreidae). *Journal of Insect Physiology*, pages 1092–1104.

Full, R. J. (1997). *Invertebrate locomotor systems*. Oxford University Press.

Full, R. J., Kubow, T., Schmitt, J., Holmes, P., and Koditschek, D. (2002). Quantifying dynamic stability and maneuverability in legged locomotion. *Integrative and comparative biology*, 42:149–157.

Full, R. J., Stokes, D. R., Ahn, A. N., and Josephson, R. K. (1998). Energy absorption during running by leg muscle in a cockroach. *The journal of experimental biology*, 201:997–1012.

Gassmann, B., Scholl, K.-U., and Berns, K. (2001). Locomotion of lauron iii in rough terrain. In *Advanced Intelligent Mechatronics, 2001. Proceedings. 2001 IEEE/ASME International Conference on*, volume 2, pages 959–964.

Graham, D. (1972). A behavioural analysis of the temporal organisation of walking movements in the 1st instar and adult stick insect *Carausius morosus*. *Journal of Comparative Physiology*, 81:23–52.

Graham, D. (1985). Pattern and control of walking in insects. *Advances in Insect Physiology*, 18:31–140.

Graham, D. and Epstein, S. (1985). Behaviour and motor output for an insect walking on a slippery surface. ii backward walking. *The Journal of Experimental Biology*, 118:287–296.

Guddat, M. and Frik, M. (2000). Control of walking machines with artificial reflexes. In *Advanced Motion of Animals and Machines, Second International Symposium on Impact and Friction of Solids, Structures and Intelligent Machines*, Montreal, Canada.

Hess, D. and Büschges, A. (1999). Role of proprioceptive signals from an insect femur-tibia joint in patterning motoneuronal activity of an adjacent leg joint. *Journal of Neurophysiology*, 81:1856–1865.

Jander, R. and Volk-Heinrichs, I. (1970). Das strauch-spezifische visuelle perceptorsystem der Stabheuschrecke (*Carausius Morosus*). *Z Vgl Physiol*, 70:425–447.

Jindrich, D. L. and Full, R. J. (1999a). Many-legged maneuverability: dynamics of turning in hexapods. *The Journal of Experimental Biology*, 202:1603–1623.

Jindrich, D. L. and Full, R. J. (1999b). Many-legged maneuverability: Dynamics of turning in hexapods. *J. Exp. Bio.*, 202:1603–1623.

Jindrich, D. L. and Full, R. J. (2002). Dynamic stabilization of rapid hexapedal locomotion. *The Journal of Experimental Biology*, 205:2803–2823.

Kimura, S., Yano, M., and Shimizu, H. (1994). A self-organizing model of walking patterns of insects ii. the loading effect and leg amputation. *Biological Cybernetics*, 70:505–512.

Kindermann, T. (2001). Behavior and adaptability of a six-legged walking system with highly distributed control. *Adapt. Behav.*, 9:16–41.

Kindermann, T., Cruse, H., and Dean, J. (1998). A biologically motivated controller for a six-legged walking system. In *Industrial Electronics Society, 1998. IECON '98. Proceedings of the 24th Annual Conference of the IEEE*, volume 4, pages 2168–2173.

Kingsley, D. A., Quinn, R. D., and Ritzmann, R. E. (2003). A cockroach inspired robot with artificial muscles. In *International Symposium on Adaptive Motion of Animals and Machines (AMAM 2003)*, Kyoto, Japan.

Kistemaker, D. A. and Soest, A. J. K. V. (2007). A model of open-loop control of equilibrium position and stiffness of the human elbow joint. *Journal of Biological Cybernetics*, pages 341–350.

Kittman, R. and Schmitz, J. (1992). Functional specialization of the scoloparia of the femoral chordotonal organ in stick insect. *The Journal of Experimental Biology*, 173:91–108.

Kittmann, R. (1997). Neural mechanism of adaptive gain control in a joint control loop: Muscle force and motoneuronal activity. *The Journal of Experimental Biology*, 200:1383–1402.

Kittmann, R., Dean, J., and Schmitz, J. (1991). An atlas of the thoracic ganglia in the stick insect, carausius morosus. *Philosophical Transactions: Biological Sciences*, 331(1260):101–121.

Lindsell, O. (2005). Distributed visual tracking of multiple mobile robots. Master's thesis, University of Edinburgh, School of Informatics. Artificial Intelligence.

Mitchell, M. (1998). *An Introduction to Genetic Algorithms*. MIT press.

Mu, L. and Ritzmann, R. E. (2005). Kinematics and motor activity during tethered walking and turning in the cockroach, *Blaberus discoidalis*. *J. Comp. Psychol.*, 191:1037–1054.

Muybridge, E. (1887). Animal locomotion.

Nair, S. S., Singh, R., Waldron, K. J., and Vohnout, V. J. (1992). Power system of a multi-legged walking robot. *Robotics and Autonomous Systems*, 9:149–163.

Nelson, G. and Quinn, R. (1998). Posture control of a cockroach-like robot. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 1, pages 157–162.

Nelson, G. and Quinn, R. (1999). Posture control of a cockroach-like robot. *Control Systems Magazine, IEEE*, 19:9–14.

Nelson, G., Quinn, R., Bachmann, R., and Flannigan, W. (1997). Design and simulation of a cockroach-like hexapod robot. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, Alburquerque, New Mexico.

Noah, J. A., Quimby, L., Frazier, S. F., and Zill, S. N. (2001). Force detection in cockroach walking reconsidered: discharges of proximal tibial campaniform sensilla when body load is altered. *The Journal of Comparative Physiology A*, 187:769–784.

Ota, Y., Inagaki, Y., Yoneda, K., and Hirose, S. (1998). Research on a six-legged walking robot with parallel mechanism. In *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, volume 1, pages 241–248.

Pearson, K. G. and Fourtner, C. R. (1975). Nonspiking interneurons in walking system of the cockroach. *Journal of Neurophysiology*, 38:33–52.

Pfeiffer, F., Eltze, J., and Weidemann, H.-J. (1995). Six-legged technical walking considering biological principles. *robotics and autonomous systems*, 14:223–232.

Pfeiffer, F., Weidemann, H.-J., and Danowski, P. (1991). Dynamics of the walking stick insect. *Control Systems Magazine, IEEE*, 11:9–13.

Poulet, J. F. A. and Hedwig, B. (2005). Auditory orientation in crickets: Pattern recognition controls reactive steering. In *Proceedings of the National Academy of Science of the United States of America*, number 43, pages 15665–15669.

Pratihar, D. K., Deb, K., and Ghosh, A. (2002). Optimal path and gait generations simultaneously of a six-legged robot using a ga-fuzzy approach. *Robotics and Autonomous Systems*, 41:1–20.

Quinn, R. D., Offi, J. T., Kingsley, D. A., and Ritzmann, R. E. (2002). Improved mobility through abstracted biological principles. In *Proceedings of the 2002 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems EPFL*, Lausanne, Switzerland.

Raibert, M. H. (1986). Legged robots. *Communication of the ACM*, 29:499–514.

Rosano, H. (2004). Six-legged biologically inspired walking machine for rough terrain. Abstract presented at Biro-net symposium.

Rosano, H. and Webb, B. (2006). The control of turning in real and simulated stick insects. In Nolfi, S., Baldassarre, G., Calabretta, R., Hallam, J., Marocco, D., Meyer, J.-A., and Parisi, D., editors, *From animals to animats 9*, volume 4095 of *Lecture Notes in Artificial Intelligence*, pages 145–156. Berlin, Germany: Springer Verlag. Proceedings of the Ninth International Conference on Simulation of Adaptive Behavior.

Rosano, H. and Webb, B. (2007). A dynamic model of thoracic differentiation for the control of turning in the stick insect. *Biological Cybernetics*, 97:229–246.

Sathya, K., Zill, S. N., Quinn, R. D., Ritzmann, R. E., and Choi, J. (2001). Finite element analysis of strain in a *Blaberus* cockroach leg during climbing. *Proceedings of the 2001 IEEE/RSI International Conference on Intelligent Robotics and Systems*, 2:833–838.

Sauer, A. E. and Stein, W. (1999a). Physiology of vibration-sensitive afferents in the femoral chordotonal organ of the stick insect. *Journal of Comparative Physiology A*, 184:253–263.

Sauer, A. E. and Stein, W. (1999b). Sensory pathways processing vibratory signals from the femoral chordotonal organ of the stick insect. *Journal of Comparative Physiology A*, 185:21–31.

Schmidt, J., Fisher, H., Haas, R., and Büschges, A. (2001). Pattern generation for walking and searching movements of a stick insect leg. ii. control of motoneuronal activity. *Journal of Neurophysiological*, 85:354–361.

Schmitt, J. and Holmes, P. (2000). Mechanical models for insect locomotion: Dynamics and stability in the horizontal plane ii. application. *Biological Cybernetics*, 83:517–527.

Schmitz, J. (1993). Load-compensating reactions in the proximal leg joints of stick insects during standing and walking. *The Journal of Experimental Biology*, 183:15–33.

Schmitz, J., Dean, J., Kindermann, T., Schumm, M., and Cruse, H. (2001). A biologically inspired controller for hexapod walking: Simple solutions by exploiting physical properties. *The biological bulletin*, 200:195–200.

Schmitz, J. and Haßfeld, G. (1989). The treading-on-tarsus reflex in stick insect: Phase-dependence and modifications of the motor output during walking. *The Journal of Experimental Biology*, 143:373–388.

Schneider, A. (2006). *Local Positive Velocity Feedback for the Movement Control of Elastic Joints in Closed Kinematic Chains*. PhD thesis, University of Bielefeld.

Schneider, A., Cruse, H., and Schmitz, J. (2005a). A bio-inspired joint controller for the decentral control of a closed kinematic chain consisting of elastic joints. In *Proceedings of the 44th IEEE Conference on Decision and Control, and*, pages 223–238.

Schneider, A., Cruse, H., and Schmitz, J. (2005b). A biologically inspired active compliant joint using local positive velocity feedback (lpvf). In *IEEE Transactions on systems, man, and cybernetics–PART B: Cybernetics*, volume 35, pages 1120–1130.

Schneider, A., Cruse, H., and Schmitz, J. (2005c). Switched local positive velocity feedback controllers: Local generation of retraction forces and inter-joint coordination during walking. In *Proceedings of the 3rd International Symposium on Adaptive Motion in Animals and Machines (AMAM 2005, on CD)*.

Schneider, A., Cruse, H., and Schmitz, J. (2006). Decentralized control of elastic limbs in closed kinematic chains. *The International Journal of Robotics Research*, 25:913–930.

Schneider, A., Fischer, B., Cruse, H., and Schmitz, J. (2007). A self-adjusting universal joint controller for standing and walking legs. In *Proceedings of the 10th International Conference on Climbing and Walking Robots (CLAWAR 2007)*.

Schumm, M. and Cruse, H. (2006). Control of swing movement: influences of differently shaped substrate. *Journal of Comparative Physiology A*, 192:1147–1164.

Stein, W. and Sauer, A. E. (1998). Modulation of sensorimotor pathways associated with gain changes in a posture-control network of an insect. *Journal of Comparative Physiology A*, 183:489–501.

Strauß, R. and Heisenberg, M. (1990). Coordination of legs during straight walking and turning in *Drosophila melanogaster*. *J. Comp. Psychol.*, pages 403–412.

Ting, L. H., Blickhan, R., and Full, R. J. (1994). Dynamic and static stability in hexapedal runners. *The Journal of Experimental Biology*, 197:251–269.

Todd, D. (1985). *Walking Machines an Introduction To Legged Robots*. Kogan Page.

Webb, B. (2001). Can robots make good models of biological behaviour? *Behavoral and Brain Sciences*, 24:1033–1050.

Webb, B. (2006). Validating robotic models. *Journal of Neural Engineering*, 3(3):R25–R35. tutorial.

Weidemann, H.-J., Eltze, J., and Pfeiffer, F. (1993a). Leg design based on biological problems. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, volume 3, pages 352–358.

Weidemann, H.-J., Pfeiffer, F., and Eltze, J. (1993b). A design concept for legged robots derived from the walking stick insect. In *Intelligent Robots and Systems '93, IROS '93. Proceedings of the 1993 IEEE/RSJ International Conference on*, volume 1, pages 545–552.

Weidemann, H.-J., Pfeiffer, F., and Eltze, J. (1994). The six-legged tum walking robot. In *Intelligent Robots and Systems '94. 'Advanced Robotic Systems and the Real World', IROS '94. Proceedings of the IEEE/RSJ/GI International Conference on*, volume 2, pages 1026–1033.

Wolf, H., Bässler, U., and Spieße, R. (2001). The femur-tibia control system in a proscopiid (caerlifera, orthoptera): A test for assumptions on the functional basis

and evolution of twig mimesis in stick insects. *The Journal of Experimental Biology*, 204:3815–3828.

Wolf, H. and Büschges, A. (1995). Nonspiking local interneurons in insect leg motor control ii. role of nonspiking local interneurons in the control of leg swing during walking. *Journal of Neurophysiology*, 73(5):1861–1875.

Zolotov, V., Fransevich, L., and Falk, E.-M. (1975). Kinematic der phototaktischen drehung bei der honigbiene *Apis mellifera* l. *J. Comp. Psychol.*, pages 339–353.